



## M.Sc. Thesis

---

# Design and Realization of a Digital Baseband Subsystem of Wakeup Receiver for Wireless Sensor Networks

Sijie Chen

### Abstract

In the development of wireless sensor networks, the lifetime of a sensor node is always a key design consideration. Since the battery in a sensor node can usually not be recharged or changed, power management is an effective way to extend the network lifetime. The wireless transceiver, also regarded as the ‘main radio’, is a relatively power hungry component in a sensor node. Therefore, an auxiliary always-on hardware ‘wakeup radio’ was proposed in order to reduce the overall power consumption. The wakeup radio listens to the wireless channel whereas the main radio is only active for a rather short time when the wakeup radio receives the packet with a certain pattern. Consequently, the power efficiency becomes a primary concern in the design of wakeup radio.

This thesis focuses on the low power design and implementation of a digital baseband subsystem in the wakeup radio. Firstly, the architecture and details of the subsystem are described. Then the design is verified by both a Spartan-3 FPGA board and TSMC90 chips. The design is functional as designed. In the end, the chip measurement setup and results are discussed. The power consumption varies from 2.1  $\mu$ w to 8.4  $\mu$ w, within our design target of 10  $\mu$ w. To our knowledge, it is the first work on the digital implementation and chip measurement of the wakeup radio.



# Design and Realization of a Digital Baseband Subsystem of Wakeup Receiver for Wireless Sensor Networks

---

THESIS

submitted in partial fulfillment of the  
requirements for the degree of

MASTER OF SCIENCE

in

MICROELECTRONICS

by

Sijie Chen  
born in Wuhan, China

Circuits and Systems Group  
Department of Microelectronics & Computer Engineering  
Faculty of Electrical Engineering, Mathematics and Computer Science  
Delft University of Technology



DELFT UNIVERSITY OF TECHNOLOGY  
DEPARTMENT OF  
MICROELECTRONICS & COMPUTER ENGINEERING

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled “**Design and realization of a digital baseband subsystem of wakeup receiver for wireless sensor networks**” by **Sijie Chen** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: Dec 10th 2009

Chairman:                      prof.dr.ir. Edoardo Charbon

Advisor:                      dr. ir. Nick van der Meijs

Committee Members:      dr.ir. G.J.M. Janssen  
                                     dr. Yan Zhang



# Abstract

---

In the development of wireless sensor networks, the lifetime of a sensor node is always a key design consideration. Since the battery in a sensor node can usually not be recharged or changed, power management is an effective way to extend the network lifetime. The wireless transceiver, also regarded as the ‘main radio’, is a relatively power hungry component in a sensor node. Therefore, an auxiliary always-on hardware ‘wakeup radio’ was proposed in order to reduce the overall power consumption. The wakeup radio listens to the wireless channel whereas the main radio is only active for a rather short time when the wakeup radio receives the packet with a certain pattern. Consequently, the power efficiency becomes a primary concern in the design of wakeup radio.

This thesis focuses on the low power design and implementation of a digital baseband subsystem in the wakeup radio. Firstly, the architecture and details of the subsystem are described. Then the design is verified by both a Spartan-3 FPGA board and TSMC90 chips. The design is functional as designed. In the end, the chip measurement setup and results are discussed. The power consumption varies from 2.1  $\mu\text{W}$  to 8.4  $\mu\text{W}$ , within our design target of 10  $\mu\text{W}$ . To our knowledge, it is the first work on the digital implementation and chip measurement of the wakeup radio.





# Acknowledgments

---

This work was taken place in Holst Centre, Eindhoven, for a cooperation project between the ultra low power wireless communication group and the ultra low power signal processing group. The project duration is from Oct 2008 to Aug 2009. During this time, many people indeed helped and encouraged me a lot.

First of all, I want to thank my supervisor Nick van der Meijs. He tried to provide me as many thesis projects as possible at the beginning, and also gave me much confidence to choose what I am most interested in. He shared a lot of experiences in scientific writing and research skills with me, which benefit me greatly. I obtained much valuable advice on my thesis as well as the attitude of being an engineer from Nick. Thank you!

I want to thank Qin Tang, who is a PhD student in CAS group. Thank you for giving me many useful comments and helping me to revise my thesis. I gained a lot from the discussions with you.

I appreciate my supervisors at Holst Centre for their great guidance and help. I am very grateful to Yan Zhang's experienced supervision. You explained the concept of the wakeup radio and help me understand the related background and technologies in the area of communication. Thank you for the cooperation in the project, for kindly answering all my questions and for your concern about my progress all the time. I have learned a lot from you. Thank Guido Dolmans for giving me this important opportunity, and for your tremendous support in the tapeout and measurement. Thank you for all the meetings and discussions with me, which were very useful and valuable in my current project and subsequent projects. I would like to thank Jos Huisken for your rich and inspiring ideas and your comments on my thesis. I also want to thank you for the laudatory "kick-off" speech that encouraged me a lot.

I would like to thank all the excellent colleagues at Holst Centre, without whom I can not finish my project successfully. I feel lucky to meet you all:

Thank Xiaoyan Wang for the kind support during the first tapeout. Thank Pieter Harpe and Xiongchuan Huang for the help in the measurement. Thank Nauman Farooq Kiyani and De Francisco Martin Ruben for the nice questions and discussions when I presented. Thank Cui Zhou, Li Huang and Dires Neirynck for the kindness and encourage.

Thank Ben Busze for the help in the second tapeout. Thank Michael de Nil for the support in the digital design flow. Thank Jun Zhou for helping me debug the timing problems in the circuit and for the patient discussions with me. Thank Yu Pu for sharing your experience in the measurement.

Thank Margot Nijkamp-Diesfeldt and Linda Oosterbosch in the HR department for the kind support and management.

Thanks are given to all my friends around me during these two years. I can not mention their names one by one so as not to miss someone, but I want to say thank you all because of the discussions, kindness and happiness.

I also would like to thank readers and committee members for reading my thesis.

Last but not least, I would like to thank my parents for their selfless love and support during all these years. This thesis is dedicated to my parents.

Sijie Chen

Delft, the Netherlands

Oct 2009

# Contents

---

Abstract	v
Acknowledgments	vii
1 Introduction .....	1
1.1 Technical background.....	1
1.2 Motivations and requirements .....	3
1.3 Related work .....	6
1.4 System description.....	6
1.5 Thesis overview .....	7
2 Basic design and implementation .....	9
2.1 Architecture .....	9
2.2 Matched filter.....	10
2.3 Synchronizer .....	16
2.4 Manchester decoder .....	16
2.5 Correlator and amplitude estimation.....	18
2.5.1 Soft-bit representation .....	18
2.5.2 Soft-bit correlation .....	18
2.5.3 Amplitude estimation .....	20
2.5.4 Simulations on threshold coefficients .....	24
2.6 Summary .....	26
3 Other implementation details .....	27
3.1 Cooperation with SPI.....	27
3.2 CSD coefficients in filter .....	29
3.3 Quadruple signal processing.....	31
3.4 Clock gating.....	33
3.5 Multiple $V_T$ optimization.....	34
3.6 Link information extraction .....	35
3.7 Summary .....	36

4 Simulation results and analysis .....	37
4.1 Testing flow .....	37
4.2 Synthesis results.....	38
4.3 Power consumption.....	38
4.4 Summary .....	39
5 Hardware implementations.....	41
5.1 FPGA verification.....	41
5.2 Tape out measurement .....	42
5.3 Summary .....	47
6 Conclusions and future work .....	49
6.1 Conclusions.....	49
6.2 Future work.....	50
Bibliography .....	51

# List of Figures

---

Fig. 1.1 WSN for a real-time traffic monitoring and automated response methodology.	1
Fig. 1.2 Multi-hop transmission in WSN .....	2
Fig. 1.3 Power management strategies in a sensor node .....	3
Fig. 1.4 Operation schedule of sensor node with duty cycling .....	4
Fig. 1.5 Operation schedule of sensor node with wakeup radio.....	5
Fig. 1.6 Handshake mechanism in wake-up scheme.....	5
Fig. 1.7 System structure of wakeup radio.....	6
Fig. 2.1 Block diagram of DBB subsystem with operating frequencies .....	9
Fig. 2.2 Basic structure of beacon packet.....	10
Fig. 2.3 Consecutive raised-cosine impulses.....	11
Fig. 2.4 The raised cosine frequency domain response.....	11
Fig. 2.5 The raised cosine time domain response.....	12
Fig. 2.6 Structure of matched filter .....	13
Fig. 2.7 Matlab script to generate reference filter coefficients .....	13
Fig. 2.8 Reference filter coefficients .....	14
Fig. 2.9 Frequency response with different filter coefficients .....	15
Fig. 2.10 Frequency response in logarithm with different filter coefficients.....	15
Fig. 2.11 Block diagram of the matched filter .....	15
Fig. 2.12 Block diagram of the synchronizer .....	16
Fig. 2.13 Example of Manchester encoding scheme showing Thomas convention.....	17
Fig. 2.14 Block diagram of the Manchester decoder .....	17
Fig. 2.15 Example of hard bit and soft bit ( $n=2$ ) representation .....	18
Fig. 2.16 Block diagram of DBB subsystem with data width.....	18
Fig. 2.17 Examples of soft-bit similarity.....	19
Fig. 2.18 Block diagram of correlator .....	20
Fig. 2.19 Relationship between $a_i$ at transmitter and $S_i$ at receiver.....	20
Fig. 2.20 Similarity between two soft-bit values .....	20
Fig. 2.21 Quantitative relationship between $a_i$ at transmitter and $S_i$ at receiver .....	21
Fig. 2.22 Beacon packet with additional amplitude estimation sequence.....	21
Fig. 2.23 Relationship between $a_7...a_1 a_0$ at transmitter and $S_7...S_1 S_0$ at receiver .....	21
Fig. 2.24 An example of $S_7...S_1 S_0$ with SNR =12 .....	23
Fig. 2.25 Block diagram of DBB subsystem with amplitude estimation.....	23
Fig. 2.26 Preamble is “11110000”, $TC=6$ .....	24
Fig. 2.27 Preamble is “11110100”, $TC=6$ .....	24
Fig. 2.28 Preamble is “11110000”, $TC=7$ .....	24
Fig. 2.29 Preamble is “11110100”, $TC=7$ .....	25
Fig. 2.30 Preamble is “11110010”, $TC=7$ .....	25
Fig. 2.31 Preamble is “11100000”, $TC=7$ .....	25
Fig. 2.32 Preamble is “11110000”, $TC=8$ .....	25
Fig. 3.1 Block diagram of SPI.....	28
Fig. 3.2 Top level cell diagram of DBB subsystem .....	29
Fig. 3.3 Conversion flow of 2's complement to CSD .....	30

Fig. 3.4 Comparison of CSD and high accuracy coefficients .....	31
Fig. 3.5 Output data of filter with four different colors indicating four groups.....	32
Fig. 3.6 Block diagram of DBB subsystem splitting into four groups.....	32
Fig. 3.7 Latch-free clock gating .....	33
Fig. 3.8 Glitches due to mismatch between En and Clock.....	33
Fig. 3.9 Latch-based clock gating .....	34
Fig. 3.10 The glitches are eliminated by the latch.....	34
Fig. 3.11 Beacon packet with additional flag and link information .....	35
Fig. 3.12 Behavior description of hard decision block.....	35
Fig. 3.13 Processing flow of link information.....	36
 Fig. 4.1 Testing flow with SPI .....	 38
 Fig. 5.1 Scheme of FPGA implementation .....	 41
Fig. 5.2 Wakeup trigger from subsystem on Spartan-3 FPGA .....	42
Fig. 5.3 Layout view of Feb tape out .....	43
Fig. 5.4 Layout view of June tape out, DBB subsystem is in the lower left ring.....	44
Fig. 5.5 Scheme of measurement system .....	44
Fig. 5.6 Measurement system in the lab.....	45
Fig. 5.7 Wakeup triggers from subsystem on chip.....	46
Fig. 5.8 Probabilities of false alarm and miss detection.....	46

# List of Tables

---

Table 1.1 Specifications of wakeup radio .....	6
Table 1.2 Design targets of digital baseband subsystem.....	6
Table 2.1 Reference filter coefficients .....	13
Table 2.2 Simplified filter coefficients.....	14
Table 2.3 Decoding rule as per Thomas convention .....	17
Table 3.1 Connections between SPI and DBB.....	28
Table 3.2 Definitions of control bits written in Po_Reg6 .....	28
Table 3.3 Look-up table of CSD conversion.....	30
Table 3.4 Conversion with coefficients.....	31
Table 4.1 Parameters for simulating transmitter and channel in Matlab.....	37
Table 4.2 Number of gates after synthesis .....	38
Table 4.3 Post-layout power consumption (1) .....	39
Table 4.4 Post-layout power consumption (2) .....	39
Table 5.1 Device utilization summary .....	42
Table 5.2 Area of June design.....	43
Table 5.3 Input and expected output in the measurement.....	45
Table 5.4 The measurement values: current and power.....	46





## 1.1 Technical background

Wireless sensor network (WSN) is a multi-disciplinary technology that is rapidly developed in recent years, contributed by advances in wireless communications, sensor technologies and digital circuit design. WSN consists of a large number of tiny sensor nodes which can observe phenomena, process data and communicate wirelessly with other nodes in close proximity. Generally speaking, each node contains sensing components, a micro-controller unit (MCU), a radio frequency (RF) transceiver and a battery. An on-board processor can be equipped as well to carry out simple computations locally and further to reduce the data volume transmitted. The nodes are distributed within a certain space to construct a ubiquitous network. Based on the collaborative work of these nodes, the physical world can be connected to the virtual world, which provides enormous opportunities for pervasive monitoring and manipulation in our lives.

WSN is particularly suitable for the occasions that wired solution is difficult or even impossible to achieve. Wireless sensor networks can be deployed in the vicinity of wired system, thus creating a complete wired and wireless measurement and control systems. In the field of healthcare, WSN let doctors monitor patients wirelessly. Patients wear wireless sensors that transmit data to a central processing point linked to the doctors' offices. These data contain information about vital signs, body functions, patient behavior and their environments. In the case of an unusual data appearance -- like a sudden spike in blood pressure or a report that an active patient has become suddenly still -- emergency medical services could be sent to the location of the patient.



Fig. 1.1 WSN for a real-time traffic monitoring and automated response methodology [1]

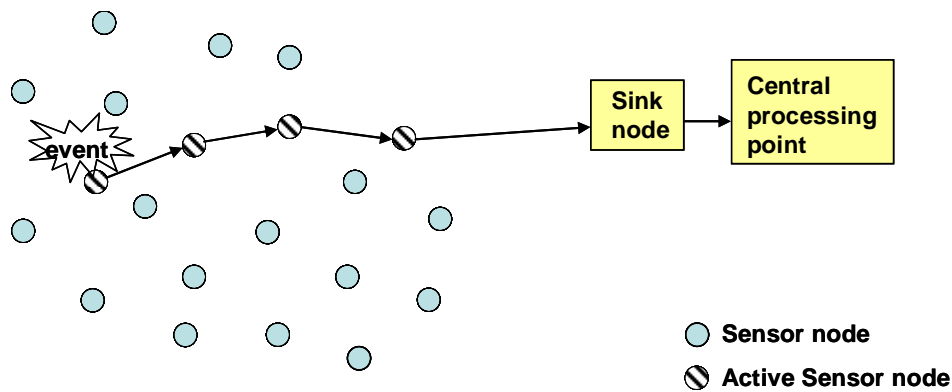


Fig. 1.2 Multi-hop transmission in WSN

WSN is an autonomous multi-hop wireless network as shown in Fig. 1.2. The sensing signal is relayed between nodes if the direct communication distance is too long. This “multi-hop” transmission reduces the transmission power required in a long distance wireless communication. The events leading to data transmission are clarified as follows: 1) the central processing point requires data from every node regularly, which is also called polling mechanism, 2) the sensor node detecting an abnormal signal will send the data immediately and automatically to the central processing point.

The wireless and distributed deployment of sensor nodes shows many advantages over traditional sensors. First of all, it is quite suitable for harsh environments since it can be installed without extra effort and infrastructures. Secondly, it is useful to attach the sensor nodes to a moving object, in order to measure the varying surrounding environment. In addition, the sensor nodes can be deployed in a region to locate signals of interest quickly and reliably, which is important in disaster relief and battlefield. Moreover, WSN could be combined with radio frequency identification (RFID) to provide wireless and seamless connectivity to remote monitoring and management system [2].

WSN provides a wide range of applications, including but not limited to the following: [1], [3]-[9].

- Precise agriculture: measurement of temperature and humidity in greenhouses;
- Home automation: security, light, energy or entertainment control;
- Environmental monitoring: contamination or structural damage detection;
- Military: reconnaissance, surveillance and targeting;
- Healthcare: medical telemetry and management;
- Logistic: object tracking and classification.

The existing wireless communication standards such as Wi-Fi and Bluetooth don't suit WSN due to the following unique requirements of WSN:

- Low power consumption
- Large scale of deployment
- Self-organizing networking
- Ability to withstand harsh environmental conditions
- Low data rate
- Low latency

Currently, the Zigbee protocol/IEEE 802.15.4 [10] is often considered as the most applicable standard for WSN, although it was not intended for this application area.

## 1.2 Motivations and requirements

Although the applications of WSN are promising and attractive, the realization is restricted by many practical factors, within which the lifetime is a key design consideration. The lifetime of WSN critically depends on the lifetime of individual sensor nodes, which are supposed to work continuously over months and even years. However, the battery in a sensor node is usually not rechargeable or changeable. In order to improve the node lifetime, two solutions can be envisioned. One is scavenging power from the environment or body skin [11] while another method is to reduce the power consumption of the node. For the sake of the given research scope, this thesis focuses on the second approach, which can be considered as power management as well.

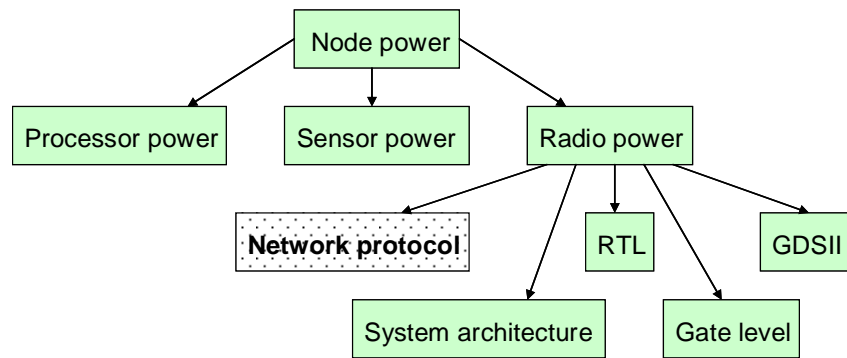


Fig. 1.3 Power management strategies in a sensor node

As shown in Fig. 1.3, the power of a sensor node is mainly dissipated in three components: the processor, the sensor, and the radio. The power consumed by wireless communications is generally the dominant part of the power consumption [12]. According to the literature [13][14], the existing WSN transmitters consume power in the range of milliwatts. To reduce the power consumption in the communication elements, the energy awareness must be taken into account in all levels as in Fig. 1.3. The improvement on higher level has greater influence on reducing power. Therefore our target is to prolong the lifetime of WSN by implementing a communication subsystem according to a power efficient network protocol.

The medium access control (MAC) protocol is used to harmonize medium access among multiple competing nodes. When designing MAC protocols, there are mainly two schemes that are suitable for WSN: the periodic listen and sleep scheme and the wake-up scheme.

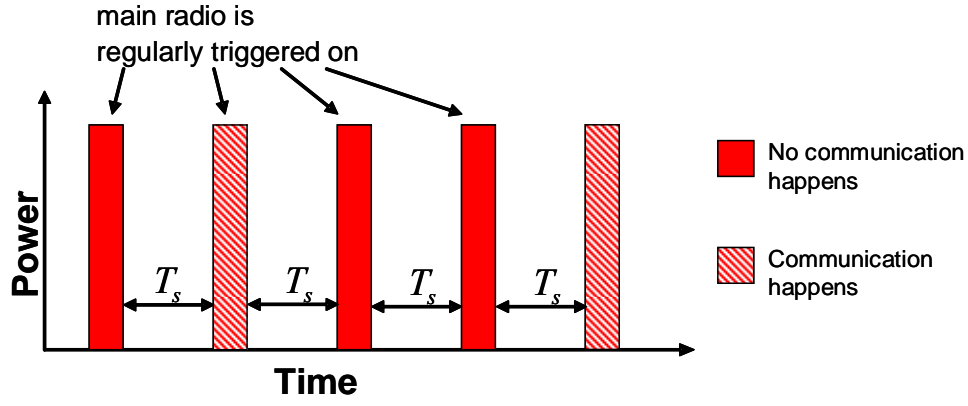


Fig. 1.4 Operation schedule of sensor node with duty cycling

In Fig. 1.4 it shows the periodic listen and sleep scheme, which is also regarded as “duty cycling”. In this scheme, the transceiver is periodically activated to listen to the channel and check whether there is a communication request to be dealt with. If so, a four-way handshake (RTS-CTS-DATA-ACK, see details in the next page) as shown in Fig. 1.5 takes place. It is also possible that no communication request is received during this period. After the active period, the transceiver is suspended for  $T_s$ , which is controlled by a timer in the MCU. During  $T_s$  the communication unit is shut down and turned to sleep mode to reduce the power consumption.

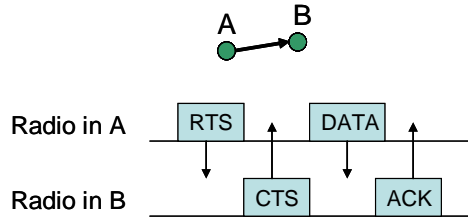


Fig. 1.5 Handshake mechanism in duty cycling scheme

The feature of this system is that the transceiver is activated every  $T_s$  period *no matter* whether a communication request arrives at it. Latency is an expression of how much time it takes for a packet from one point to another. In such a system with regular schedule, power and latency is a trade-off. Low power is achieved by increasing  $T_s$  to activate the transceiver less frequently. However, the price is longer latency because there is larger possibility that the communication request is missed during the longer inactive period and retransmission is needed. What is more, a node must synchronize itself with its neighbor to listen and sleep at the same time. When a channel is established between two nodes, the surrounding nodes need to enter sleep mode to avoid collision, and further the synchronization within the whole network needs to be updated all the time. In other words, such a protocol is complicated.

By contrast to the listen and sleep scheme, the wake-up scheme can satisfy low power and low latency at the same time. The idea was at first proposed in a cell phone system integrating a paging system in [15]. In order to keep the transceiver in sleep mode for most of the time and activate it *only* when a communication request is received, an auxiliary hardware receiver was proposed. In such a dual-radio system, the transceiver is called the “main radio”, while the auxiliary receiver is called the “wake-up (Wu) radio”. The wakeup radio listens to the channel continuously and

releases a wakeup signal to the main radio *immediately as* the packet with a certain pattern is received. The main radio will switch from the sleep mode to the active mode after the wakeup signal is present and fall into the sleep mode again after the communication is terminated, as shown in Fig. 1.6. Therefore the main radio is only active for a relatively short time and the wakeup radio is always on to capture every communication request with small latency.

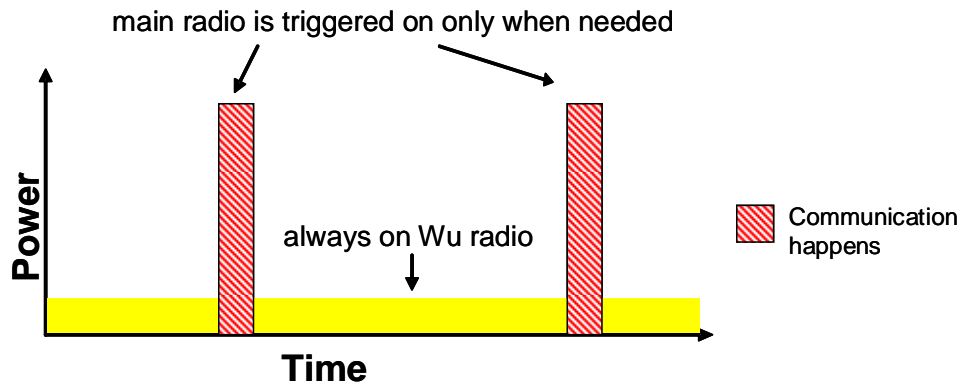


Fig. 1.6 Operation schedule of sensor node with wakeup radio

If the wakeup radio can be sustained with ultra-low power consumption, the total power will be reduced significantly. This is possible by constructing a radio with simple architecture. In addition, we can benefit from the wake-up scheme because of the simplified protocol as well. A node in such a network doesn't need to know the status of its neighbors.

The communication between nodes is guaranteed by a handshake mechanism, which is common in data transmission system. As shown in Fig. 1.7, it is a RTS-WU-CTS-DATA-ACK five-way handshake compared to the ordinary four-way one [16]. When a node needs to transmit data, firstly it will release 'request to send (RTS)' to the target node. When the RTS is captured by the wakeup radio in the target node, the main radio is triggered on and sends back a 'cleared to send (CTS)' signal. Then the data channel is established and will be ended up with the acknowledgment (ACK) signal if the transmission is completed.

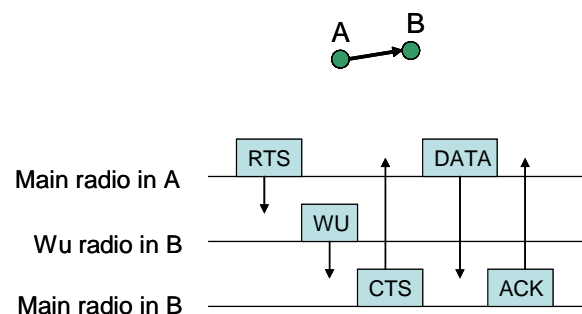


Fig. 1.7 Handshake mechanism in wake-up scheme

There are many sensor nodes in a network. However, not all of them need to be woken up at one time. So an address sequence is adopted thus each node has a unique local address and each beacon packet contains only one address. In this way, it can be guaranteed that one packet being transmitted will only wake up one node at a time.

In order to show a general impression on how the wakeup radio behaviors, the overview of the wakeup radio proposal is depicted in Table 1.1. The descriptions are given by ultra low power (ULP) wireless communication group at Holst Centre.

Table 1.1 System descriptions of wakeup radio

Power consumption	$< 50 \mu\text{w}$
Communication range	$\approx 10 \text{ m}$
Data rate	200 kbps
Carrier frequency	2.4 GHz
Signal modulation	On-off-keying (OOK)

It can be seen from the previous description that the wakeup radio is an always-on component in every sensor node. Hence the primary requirement of the wakeup radio should be low power. This thesis focuses on the design and implementation of the digital baseband (DBB) subsystem in the wakeup radio. We have the following design specifications as in Table 1.2, which is also provided by ULP wireless communication group:

Table 1.2 Design specifications of digital baseband subsystem

Power consumption	$< 10 \mu\text{w}$
Area	$< 1 \text{ mm}^2$
Data rate	200 kbps
Signal-to-noise ratio (SNR)	$> 8 \text{ dB}$
Length of beacon packet	$< 0.3 \text{ ms}$

## 1.3 Related work

In [17] and [18], a three-stage wakeup scheme implemented in BiCMOS technology was proposed. Zarlink semiconductor provided a commercial medical implant communication system (MICS) where an ultra low power wakeup receiver was applied [19]. In [20] it was discussed that a low-cost wakeup radio made out of standard components for the use in the 868 MHz band. In [21] the architecture of a wakeup receiver was discussed, yet the digital baseband part of which was implemented off-line in Matlab. To our knowledge, neither the work on the detailed description nor on the hardware implementation of the digital baseband part of the wakeup receiver can be found in literature.

## 1.4 System description

The system structure of our wakeup radio proposal is as described in Fig. 1.8.

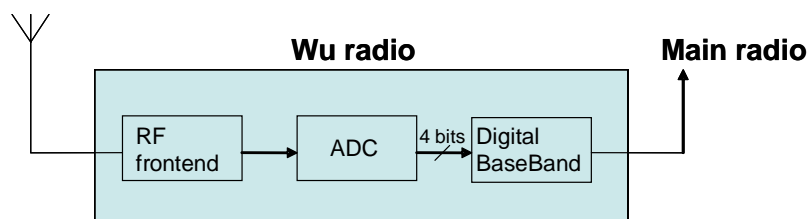


Fig. 1.8 System structure of wakeup radio

The RF front-end circuit consists of an RF amplifier, an envelope detector, baseband amplification stages, and double-sampling circuitry. The sampling clock signal is at the frequency of 20 MHz.

The analog-to-digital converter (ADC), operating at 800 KHz, connects the RF frontend and digital baseband. The output of the ADC, which is 4-bit long in soft-bit representation (see Section 2.5), feeds into the digital baseband subsystem. The resolution of the ADC is a trade-off between accuracy and power, thus 4-bit is chosen to compromise it.

## **1.5 Thesis overview**

The rest of the thesis is structured as follows.

In Chapter 2, the architecture overview of the digital baseband subsystem is presented. The processing flow, including the matched filter, synchronizer, decoder and correlator is discussed separately. The explanations of implementation choices are provided as well. The necessity of the amplitude estimation is discussed and the equations of the softbit correlation and threshold coefficient are deduced.

In Chapter 3, it explains several strategies in order to make the subsystem work effectively with low dynamic and leakage power. The peripheral equipment is also shown.

The subsystem is implemented in VHDL, and the testing flow and simulation results are presented and analyzed in Chapter 4. Due to the low data rate, the timing target is easily reached, so there is no timing issue.

The design is verified both by Spartan-3 FPGA and TSMC90 chips. In Chapter 5, the hardware realization, chip measurement setup and results are illustrated. The design is functional as designed. The minimal active power achieved is 2.1  $\mu$ w at the data rate 200 kbps.

Chapter 6 summaries the work achieved and provides recommendations for work in the future.





# Basic design and implementation

# 2

In the previous chapter, the background, motivation and related research of the wakeup radio have been discussed. The basic structure and processing procedure of digital baseband subsystem in the wakeup receiver is the main topic of this chapter.

The reminder of this chapter is organized as follows. The architecture overview is investigated in Section 2.1. The function blocks are described separately in Section 2.2 – 2.5, among which the explanation of the amplitude estimation is provided in Section 2.5. This chapter is summarized in Section 2.6.

## 2.1 Architecture

As mentioned in the introduction part, the wake-up scheme is such a strategy that the main radio is kept in the sleep mode when there is no data communication, and it is activated only when a communication request is received by the wakeup radio. In this way, the main radio is only active for a relatively short time whereas the wakeup radio is always on. If the wakeup radio can be sustained with ultra-low power consumption, the total power will be reduced significantly.

First of all, the block diagram of the digital baseband subsystem is presented in Fig. 2.1. It is a cascade processing flow consisting of four blocks. The 4-bit input is obtained from the ADC and goes through a matched filter first. The second block is used to synchronize the input signal with the subsystem. When the synchronization is done, the rest of the input is decoded in the Manchester decoder (see Section 2.4) and then the correlation is calculated in the correlator. Afterwards, the result is compared with a threshold. In the end, a 1-bit trigger signal is produced if the result is larger than the threshold. When the main radio is triggered on, data communications will start meanwhile the wakeup radio will reset itself to wait for the next communication request.

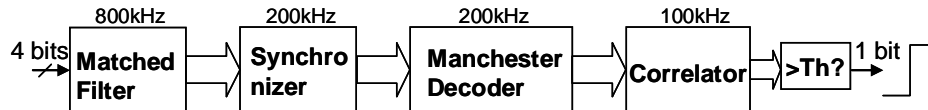


Fig. 2.1 Block diagram of DBB subsystem with operating frequencies

The data rate is 200 kbps. The oversampling ratio of  $k$  in the ADC can improve the SNR by  $10\log_{10}(k)$  [22], while higher frequency leads to higher power consumption. In order to compromise the SNR and power, 4-time sampling is chosen. Therefore the external clock is 800 kHz. The data is downsampled after the matched filter, thus a slower clock (200 kHz) is derived. Later on, an even slower clock (100 kHz) is

produced during Manchester decoding. As a result, the operating frequency of the last block is 100 kHz as shown in Fig. 2.1.

The beacon packet is a kind of small packet targeting to certain nodes with the attempt to establish a wireless connection. The basic structure of the beacon packet is shown in Fig. 2.2. It consists of two parts: the synchronization preamble and the wakeup address.



Fig. 2.2 Basic structure of beacon packet

The synchronization is necessary because the starting time of each symbol has to be determined in order to make right decisions. The synchronization preamble is composed of 8-bit. It can be seen that the presence of ‘1’ causes more power while one can benefit from the accumulated energy peak of continuous ‘1’s. To achieve a balance, the pattern of the preamble is fixed as “11110000” for any packet.

The wakeup address is a Manchester encoded signal, which is a common digital modulation scheme to enhance robustness. For every sensor node in the network, an 8-symbol-long local address is assigned to it. If the wakeup address in the beacon packet matches the local address, a trigger signal will be given and the main radio switches from the sleep mode to the active mode. The local address in each sensor node is unique in the network, which means, a beacon packet being transmitted will only wakeup one radio at a time.

## 2.2 Matched filter

Raised cosine filter is chosen to reduce the inter-symbol interference (ISI) in the signal sequence. In this design, the finite impulse response (FIR) filter is referred, because of its stability and simplicity in realization.

The raised cosine filter is such a widely used low-pass filter that produces time domain ripples that cross through zero at the midpoint of adjacent pulses [23] (see Fig. 2.3). Its frequency response is given by [23]

$$H(\omega) = \begin{cases} \tau, & 0 \leq \omega \leq c \\ \tau \left\{ \cos^2 \left[ \frac{\tau(\omega - c)}{4\alpha} \right] \right\}, & c \leq \omega \leq d \\ 0, & \omega > d \end{cases} \quad (2.1)$$

where  $\omega$  is radian frequency,  $\tau$  is the pulse period ( $\tau = T_0 = 1 / f_0$ ),  $\alpha$  is the roll off factor,  $c$  is equal to  $\pi(1 - \alpha) / \tau$ , and  $d$  is equal to  $\pi(1 + \alpha) / \tau$ .

The impulse response of the raised cosine filter is given by [23]

$$h(t) = \left( \frac{\sin\left(\frac{\pi t}{\tau}\right)}{\frac{\pi t}{\tau}} \right) \times \left( \frac{\cos\left(\frac{\pi \alpha t}{\tau}\right)}{1 - \left(\frac{2\alpha t}{\tau}\right)^2} \right) \quad (2.2)$$

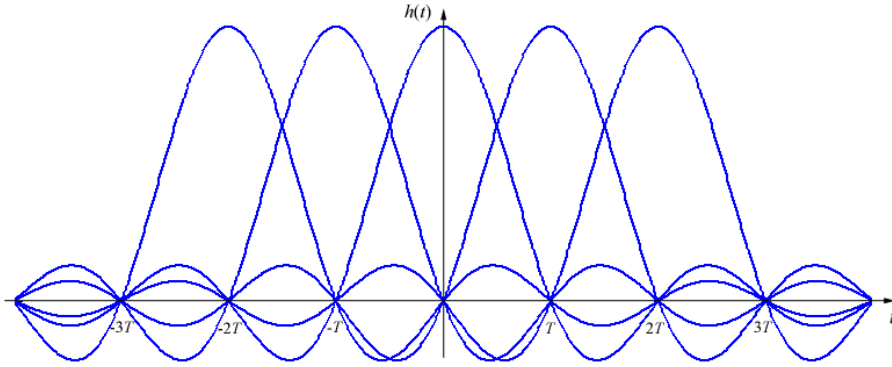


Fig. 2.3 Consecutive raised-cosine impulses [24]

As we can see from Fig. 2.4 and Fig. 2.5, the pulse shaping limits the bandwidth in frequency domain, whereas it increases the ISI between pulses in time domain. The choice of the roll-off factor  $\alpha$  is a trade-off between the bandwidth and ripple decay rate. Since the pulse is possibly not sampled exactly at the midpoint,  $\alpha = 1$  is applied to reduce the interference from adjacent pulses.

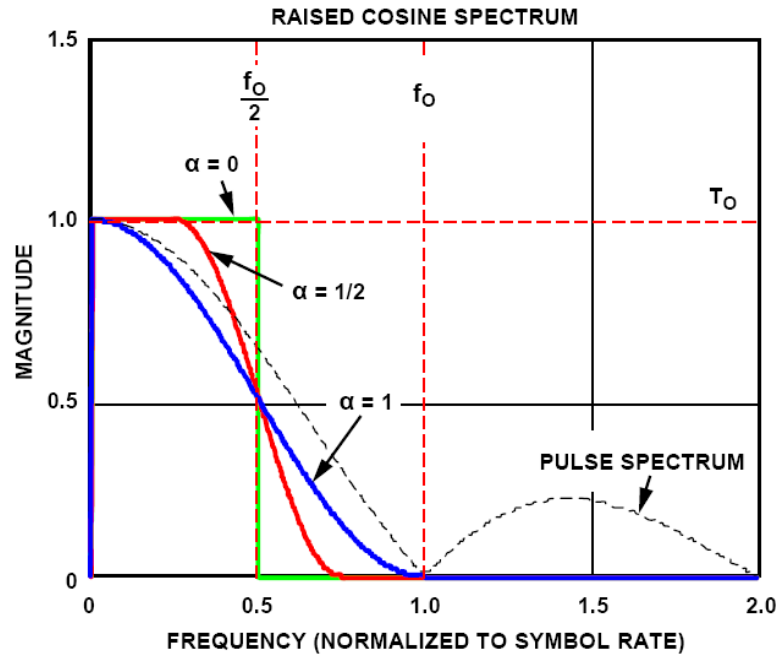


Fig. 2.4 The raised cosine frequency domain response [23]

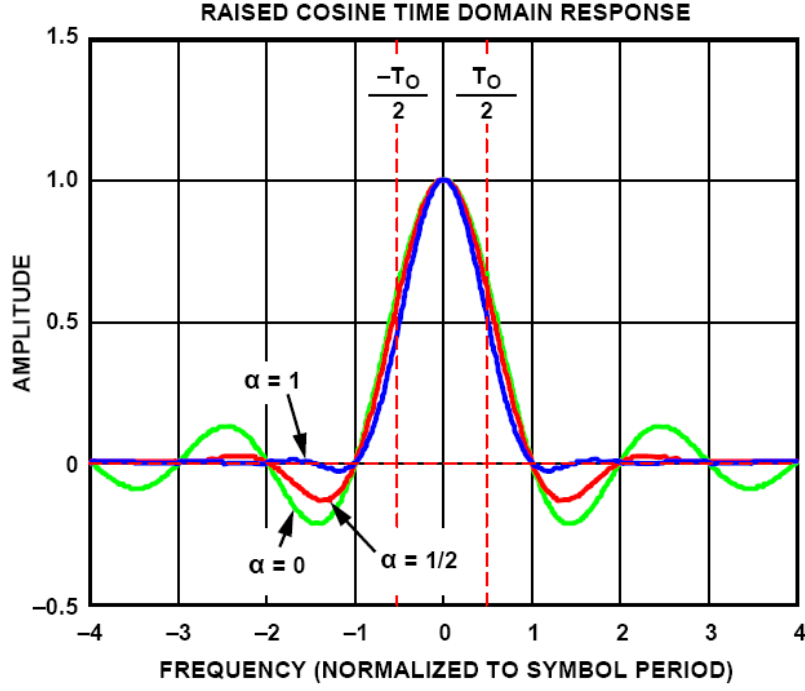


Fig. 2.5 The raised cosine time domain response [23]

The raised cosine filter can be divided into two identical root-raised-cosine (RRC) filters, whose product of frequency response yields the desired raised cosine response [23]. The filters are applied in the transmitter  $H_T(\omega)$  and receiver  $H_R(\omega)$  as the pulse shaping filter and matched filter respectively. The cooperation of two RRC filters reduces the bandwidth and ISI of signals at the same time.

$$H_T(\omega) = H_R(\omega) = \sqrt{H(\omega)} \quad (2.3)$$

The frequency response of the RRC filter is given by [23]

$$H_T(\omega) = H_R(\omega) = \begin{cases} \sqrt{\tau}, & 0 \leq \omega \leq c \\ \sqrt{\tau} \left\{ \cos \left[ \frac{\tau(\omega - c)}{4\alpha} \right] \right\}, & c \leq \omega \leq d \\ 0, & \omega > d \end{cases} \quad (2.4)$$

The structure of a matched filter is as follows:

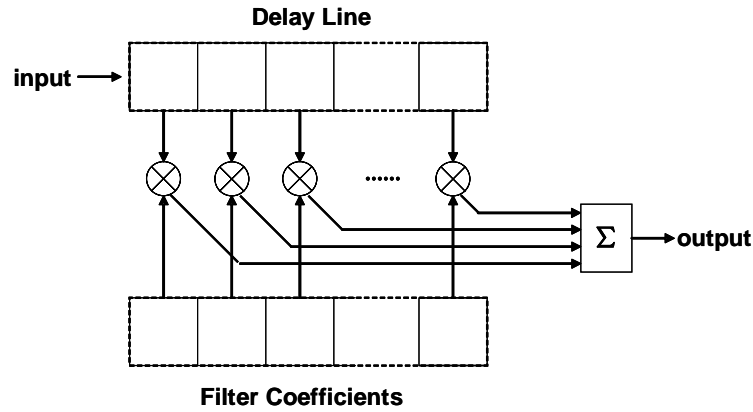


Fig. 2.6 Structure of matched filter

The Matlab code in Fig. 2.7 is used to generate the *reference* coefficients of a root-raised-cosine filter with a *filter order* 24. The result is illustrated in Table 2.1 and Fig. 2.8. *Rcosine* is a function in the Matlab Communications Toolbox. The shape of the impulse response is determined by the *nsamp* (number of samples) and *rolloff* factor. The *filter order* only influences the expansion of the impulse response. Larger order leads to longer ripple in the side lobes.

```

nsamp = 4;
filtorder = 24;
delay = filtorder / (nsamp * 2);
rolloff = 1;
rrcfilter = rcosine (1,nsamp, 'fir/sqrt',
rolloff, delay);

```

Fig. 2.7 Matlab script to generate reference filter coefficients

The number of taps in a filter is equal to the *filter order* plus one. The principle on filter order selection is minimizing the number of taps under certain filtering requirements. In the above Matlab script, *delay* is the number of samples between the filter's initial response and its peak response, so it must be an integer. Therefore we can see the minimum *filter order* applicable is 8. Then the filter coefficients are referred to the values of tap9 – tap17 in Table 2.1.

Table 2.1 Reference filter coefficients

	tap 1	tap 2	tap 3	tap 4	tap 5
decimal	-0.0045	0.0000	0.0064	0.0000	-0.0101
	tap 6	tap 7	tap 8	tap 9	tap 10
decimal	0.0000	0.0182	-0.0000	-0.0424	0.0000
	tap 11	tap 12	tap 13	tap 14	tap 15
decimal	0.2122	0.5	0.6366	0.5	0.2122
	tap 16	tap 17	tap 18	tap 19	tap 20
decimal	0.0000	-0.0424	-0.0000	0.0182	0.0000
	tap 21	tap 22	tap 23	tap 24	tap 25
decimal	-0.0101	0.0000	0.0064	0.0000	-0.0045

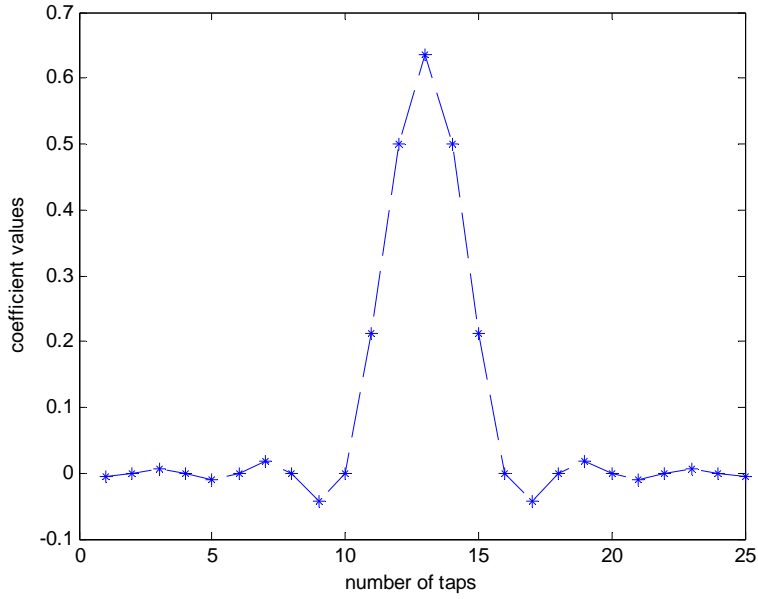


Fig. 2.8 Reference filter coefficients

When we compare the values in Table 2.1 and the shape in Fig. 2.8, we can see the large values happen at the middle range of the impulse response, which is also named as main lobe. The largest value (tap 9 or tap 17 in Table 2.1) in side lobes is approximately 5 times less than the smallest coefficient (tap 11 or tap 15 in Table 2.1) in main lobe. Therefore we try to keep these 5 coefficients in the main lobe (tap 11 - tap 15 in Table 2.1) to construct a simplified RRC filter with only 5 taps. We have the following coefficient values as in Table 2.2.

Table 2.2 Simplified filter coefficients

	tap 1	tap 2	tap 3	tap 4	tap 5
decimal	0.2122	0.5	0.6366	0.5	0.2122

The performance of the simplified coefficients in the frequency domain is also illustrated in Matlab. Since pulse shaping is an action in time domain, so the raised cosine filter is a *time domain filter*, the frequency response is of little concern [25]. When  $\alpha = 1$  is applied, the expected cutoff frequency that can be seen from Fig. 2.4 is  $f_0 = 1/T_0 = 200$  kHz. As shown in Fig. 2.9 and Fig. 2.10, the requirement of cutoff frequency is satisfied, and stop-band attenuation (-30dB) is acceptable if the simplified coefficients are chosen. Compared to the filters of order 24 and order 8, the frequency performance of the simplified filter is not deteriorated beyond the filtering requirements. The advantage is that the minimized hardware is gained to save computational power.

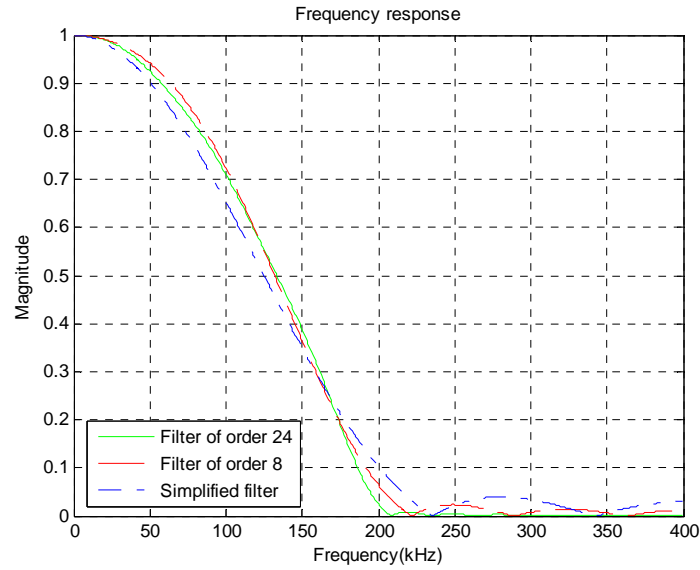


Fig. 2.9 Frequency response with different filter coefficients

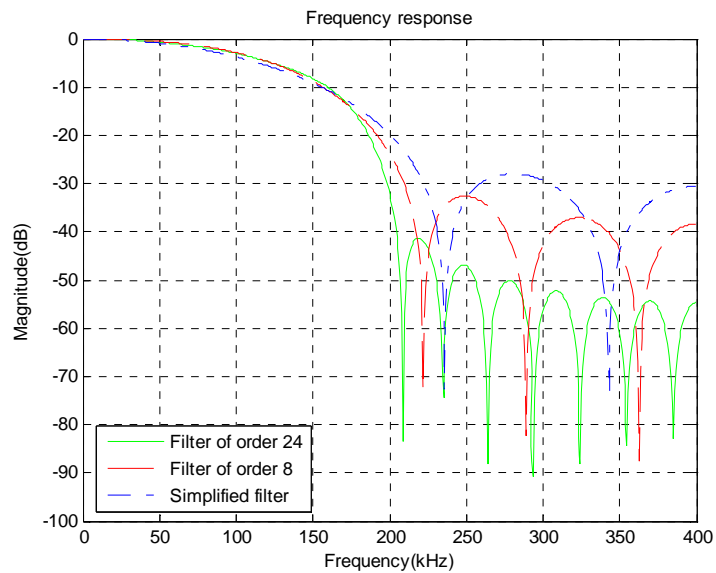


Fig. 2.10 Frequency response in logarithm with different filter coefficients

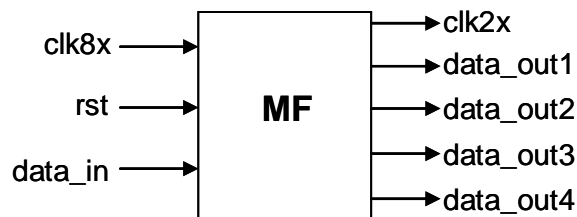


Fig. 2.11 Block diagram of the matched filter

The filter is an always on component which interprets the input data. The working clock (*clk8x*) frequency is 800 kHz. With each rising edge of the clock, the input data (*data\_in*) is shifted in and multiplied with the coefficients of the filter at each tap. The

sum of products is split into four groups (see Section 3.3) and carried out parallel as filtering outputs (*data\_out1,2,3,4*). The clock is also four times slowed down, producing a 200 kHz clock (*clk2x*).

## 2.3 Synchronizer

Without the synchronization block, the wakeup receiver can not distinguish the existence of payload from the whole beacon sequence. Because of this feature, the synchronizer is located in front of both the decoder and correlator.

The synchronizer only deals with the first part of beacon, i.e., the amplitude estimation sequence and the synchronization preamble, among which the amplitude estimation sequence is not mentioned in Fig. 2.2. The amplitude estimation is used for signal decision and will be discussed in details in Section 2.5.

The principle of the synchronizer is similar to that of the correlator, where the synchronization preamble (*data\_in*) is compared with the local one to get a distance between them. If the distance of a synchronization preamble and the predefined preamble “11110000” is within a small range, it indicates that the preamble is detected. A threshold (*p\_threshold*) is applied to decide whether the distance is close enough to confirm the detection.

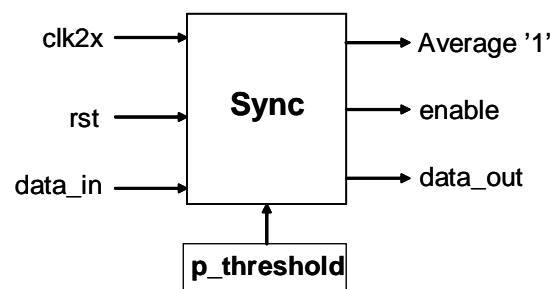


Fig. 2.12 Block diagram of the synchronizer

The synchronizer works at a 4-time slower clock (*clk2x*) derived from the external one (*clk8x*). When the circuit is synchronized with the beacon packet, the payload (*data\_out*), the output of amplitude estimation (*Average '1'*, see Subsection 2.5.3) and enable signal (*enable*) are sent to the next block.

## 2.4 Manchester decoder

The wakeup address sequence received is a Manchester encoded signal. Manchester code is a digital modulation code in data transmission systems. Each symbol has a constant bit period where one transition exists at the midpoint of each bit. Bit ‘1’ is expressed by a high-to-low transition, bit ‘0’ by low-to-high transition (according to Thomas convention) [26]. The encoding scheme is illustrated in Fig. 2.13.



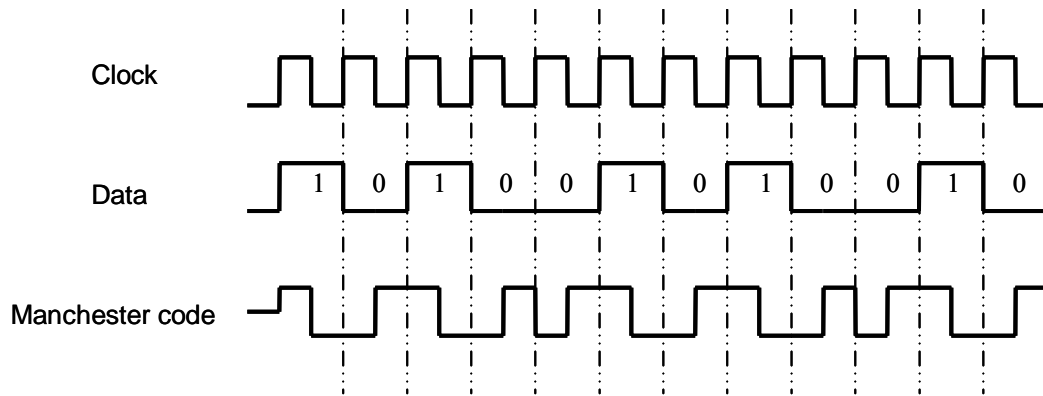


Fig. 2.13 Example of Manchester encoding scheme showing Thomas convention

From Table 2.3, we can see Manchester code as to Thomas convention is the result of the exclusive NOR logic of the clock and data.

Table 2.3 Decoding rule as per Thomas convention

Manchester code	clock	original data
1	1	1
0	0	1
0	1	0
1	0	0

Manchester code has the following advantages over the non-return-to-zero (NRZ) code [26] :

- The coded signal can be self-synchronized, since a clock period is twice as the minimum period between two transitions in Manchester code.
- There is no DC component in Manchester code so the attenuation by AC coupling can be avoided.
- Manchester code is robustness since an error bit without transition at the midpoint can be detected easily.

It also has two minor disadvantages: Manchester code requires twice as much bandwidth as the NRZ code. In addition, it needs complex decoding circuitry if the clock is unknown [26], but it is not a problem in this design provided the clock signal is presented.

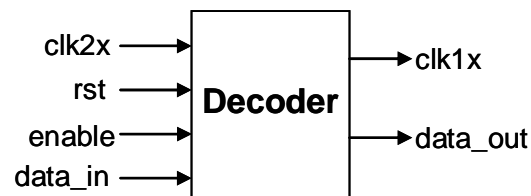


Fig. 2.14 Block diagram of the Manchester decoder

The block diagram of the Manchester decoder is shown in Fig. 2.14. It carries the first half of each input symbol (*data\_in*) to the output (*data\_out*) if the block is enabled (*enable*). In the future design, the second half of symbol can be put to use. For

example, the two parts are compared to decide the original data, and meanwhile the error bits without transitions at the midpoint can be detected.

A slower clock ( $clk1x$ ) is derived from  $clk2x$  at the same time of enabling. The output is presented with every rising edge of  $clk1x$ .

## 2.5 Correlator and amplitude estimation

### 2.5.1 Soft-bit representation

Before the soft-bit correlator is introduced, we would like to explain the hard bit and soft bit representation first.

As we all know, bit ‘1’ is assigned to high voltage ( $V \geq \frac{1}{2} V_{DD}$ ) in digital circuits, and ‘0’ indicates low voltage ( $V < \frac{1}{2} V_{DD}$ ). However, besides describing voltage amplitude in one bit, we can reorganize the amplitude into  $2^n$  small intervals by using  $n$  bits as in Fig. 2.15. The description in one bit is defined as hard bit and in multiple bits is called soft bit representation. For a Gaussian channel with additive white noise (AWGN), the additional information provided by the soft bits in most instances can provide about 2 dB of additional coding gain [27].

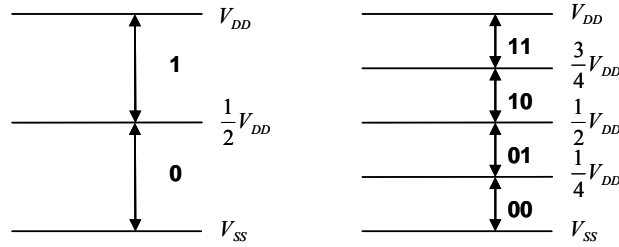


Fig. 2.15 Example of hard bit and soft bit ( $n=2$ ) representation

As in Fig. 2.16 the ADC provides a 4-bit output to describe the amplitude of the data in soft bit representation. After the matched filter, the data length is extended to 11 bits due to the fact that multiplication has taken place. In order to reduce the number of bits to save power, three least significant bits (LSBs) are truncated and the most significant eight bits are retained. For the rest of the blocks, the data width remains as 8-bit, except at the output of correlator. The correlation result is the sum of eight 8-bit addends, so it is expended to 11 bits.



Fig. 2.16 Block diagram of DBB subsystem with data width

### 2.5.2 Soft-bit correlation

The correlation, which is used to measure the similarity between two different signals quantitatively. It is obvious that the larger correlation indicates closer distance

between two signals. For hard bits ‘0’ and ‘1’, the correlation is 0, coming from  $0 \odot 1 = 0$ . The operator symbol ‘ $\odot$ ’ stands for exclusive NOR logic. For ‘1’ and ‘1’,  $1 \odot 1 = 1$  is obtained, which is to say the correlation value is 1. If we define  $i_{th}$  hard bit correlation as  $C_{hard,i}$  then we have

$$C_{hard,i} = a_i \odot b_i \quad (2.5)$$

Now we come to the correlation of two sequences in hard-bit  $a_7a_6a_5a_4a_3a_2a_1a_0$  and  $b_7b_6b_5b_4b_3b_2b_1b_0$ . The correlation  $C_{hard}$  is the accumulation of the correlations of all single bits.

$$C_{hard} = \sum_{i=0}^7 C_{hard,i} = \sum_{i=0}^7 a_i \odot b_i \quad (2.6)$$

For example, the correlation of sequences “11101100” and “11001100” is 7, as they are very similar. Sequences “00000001” and “11111111” are not so close to each other and their correlation is as low as 1.

Here, the soft-bit representation is utilized. Every hard bit  $a_i$  is replaced by 8 soft bits  $a_{i,7}a_{i,6}a_{i,5}a_{i,4}a_{i,3}a_{i,2}a_{i,1}a_{i,0}$  ( $n = 8$ ). There are  $2^8 = 256$  voltage intervals describing the exact amplitudes of  $a_i$ . To calculate the correlation between the signal  $a_{i,7} \dots a_{i,1}a_{i,0}$  and the maximum (“11111111”) or minimum (“00000000”) voltage level, Fig. 2.17 is illustrated.

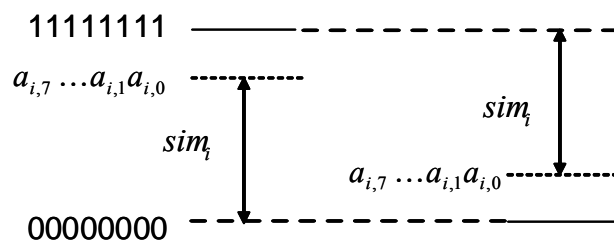


Fig. 2.17 Examples of soft-bit similarity

Fig. 2.17 proves that distance between input  $a_{i,7} \dots a_{i,1}a_{i,0}$  the opposite level of the maximum or minimum voltage level can be used to describe the similarity relationship. For instance the greater distance between  $a_{i,7} \dots a_{i,1}a_{i,0}$  and “00000000”, indicates the closer it to “11111111”. In order to calculate the distance, subtraction is applied. Therefore we have the equations for soft-bit similarity:

$$sim_i = a_{i,7} \dots a_{i,1}a_{i,0} - 00000000 \quad (2.7)$$

or

$$sim_i = 11111111 - a_{i,7} \dots a_{i,1}a_{i,0} \quad (2.8)$$

In this way, the correlation of soft-bit sequence is the accumulation of all these similarity values, where high similarities in all 8 bits lead to high correlation result.

$$C_{soft} = \sum_{i=0}^7 sim_i \quad (2.9)$$

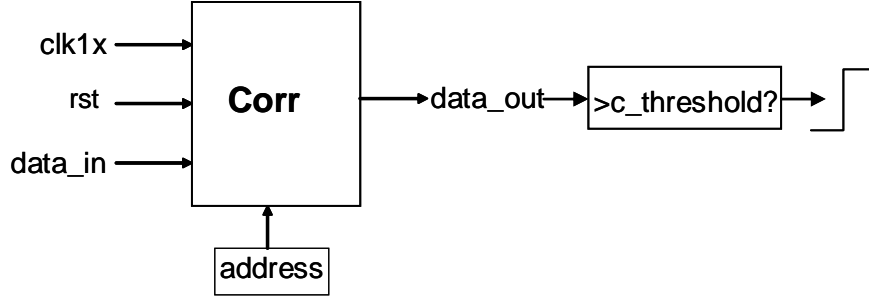


Fig. 2.18 Block diagram of correlator

Fig. 2.18 is the diagram of correlator in the design. In every clock period ( $clk1x$ ),  $data\_in$  enter this block. By comparing to the local  $address$ , a similarity value is obtained. After eight clock periods, eight similarity values are accumulated. Hereafter  $data\_out$  is produced. If  $data\_out > c\_threshold$ , a trigger signal will be generated at the output of the subsystem to wake up the main radio.

### 2.5.3 Amplitude estimation

From the previous subsection we know that in the synchronizer and correlator, the distance between the signal  $a_{i,7} \dots a_{i,1} a_{i,0}$  and the opposite level of the maximum or minimum signal level can be used to describe the similarity relationship. Here we define  $S_i$  at the receiver end is corresponding to  $a_i$  at the transmitter end.  $S_i$  is in soft-bit representation, where  $S_i = s_{i,7} s_{i,6} s_{i,5} s_{i,4} s_{i,3} s_{i,2} s_{i,1} s_{i,0}$ .  $b_i$  is the address bit at the corresponding position.



Fig. 2.19 Relationship between  $a_i$  at transmitter and  $S_i$  at receiver

As shown in Fig. 2.20, the maximum level is replaced with an actual level  $h_{i,7} h_{i,6} h_{i,5} h_{i,4} h_{i,3} h_{i,2} h_{i,1} h_{i,0}$  which is referred to the high voltage, and the minimum level is replaced with  $l_{i,7} l_{i,6} l_{i,5} l_{i,4} l_{i,3} l_{i,2} l_{i,1} l_{i,0}$  representing the low voltage level.

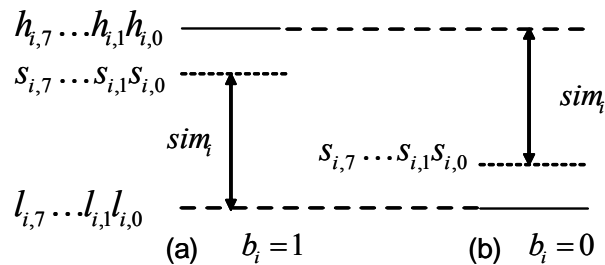


Fig. 2.20 Similarity between two soft-bit values

Consequently, we have

$$sim_i = s_{i,7} \dots s_{i,1} s_{i,0} - l_{i,7} \dots l_{i,1} l_{i,0} \quad \text{when } b_i = 1 \quad (2.10)$$

$$sim_i = h_{i,7} \dots h_{i,1} h_{i,0} - s_{i,7} \dots s_{i,1} s_{i,0} \quad \text{when } b_i = 0 \quad (2.11)$$

Therefore we come up with a question: what is the optimal value of  $h_{i,7} \dots h_{i,1} h_{i,0}$  and  $l_{i,7} \dots l_{i,1} l_{i,0}$ ? For instance, is  $h_{i,7} \dots h_{i,1} h_{i,0} = 11111111$  and  $l_{i,7} \dots l_{i,1} l_{i,0} = 00000000$  a tenable assumption?

Obviously  $h_{i,7} \dots h_{i,1} h_{i,0}$  and  $l_{i,7} \dots l_{i,1} l_{i,0}$  are not independent of the data received by the subsystem. The amplitude of  $h_{i,7} \dots h_{i,1} h_{i,0}$  should be fixed to the voltage level of corresponding signal  $S_i|_{a_i=1}$  as close as possible. Similarly,  $l_{i,7} \dots l_{i,1} l_{i,0}$  need to be close to  $S_i|_{a_i=0}$ .

We can now look into the output of the matched filter and find out that amplitude of  $S_i|_{a_i=0}$  is slightly more than “0...00”, while  $S_i|_{a_i=1}$  is much less than “1...11”. To explain this, the noisy wireless channel is taken into account. Another reason is that, in the envelop detector, the squares of data are extracted. The square of a positive number between 0 and 1 is closer to 0, but further from 1. Hence “11111111” is not a reliable representation for  $h_{i,7} \dots h_{i,1} h_{i,0}$ .

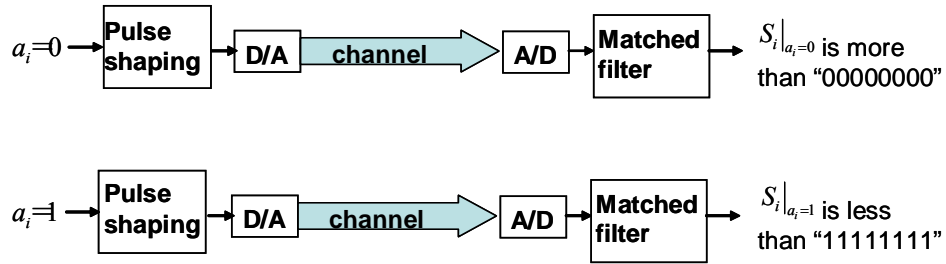


Fig. 2.21 Quantitative relationship between  $a_i$  at transmitter and  $S_i$  at receiver

In order to estimate the amplitude of  $S_i$  at the receiver end, the amplitude estimation is necessary. A fixed sequence “10101010” is transmitted as the first part of the beacon packet.

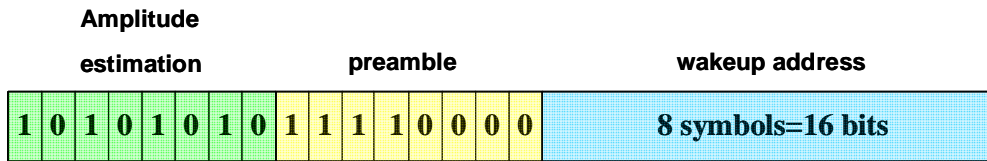


Fig. 2.22 Beacon packet with additional amplitude estimation sequence

As a result,

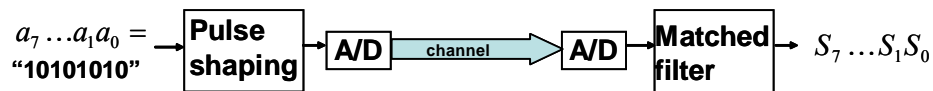


Fig. 2.23 Relationship between  $a_7 \dots a_1 a_0$  at transmitter and  $S_7 \dots S_1 S_0$  at receiver

From  $S_7 \dots S_1 S_0$  at the receiver, the following amplitudes can be known:

$$\text{Average '}\frac{1}{2}\text{' = } \frac{(S_7 + \dots + S_1 + S_0)}{8} \quad (2.12)$$

$$\begin{aligned} \text{Average '0' = } l_{i,7} \dots l_{i,1} l_{i,0} \\ = \frac{S_6 + S_4 + S_2 + S_0}{4} \end{aligned} \quad (2.13)$$

$$\begin{aligned} \text{Average '1' = } h_{i,7} \dots h_{i,1} h_{i,0} \\ = \frac{S_7 + S_5 + S_3 + S_1}{4} \end{aligned} \quad (2.14)$$

where Average ' $\frac{1}{2}$ ' is the average value of all  $S_i$ , Average '0' is the average value of  $S_i|_{a_i=0}$ , Average '1' is the average value of  $S_i|_{a_i=1}$ . Average '0' and Average '1' are the signal level  $l_{i,7} \dots l_{i,1} l_{i,0}$  and  $h_{i,7} \dots h_{i,1} h_{i,0}$  required in the correlation. However, Average '0' and Average '1' can not be obtained directly in the realization because the amplitude estimation is an operation before synchronization. So Average ' $\frac{1}{2}$ ' is used in calculation.

We assume

$$\text{Average '0' = } l_{i,7} \dots l_{i,1} l_{i,0} \approx 00000000 \quad (2.15)$$

Therefore from (2.12)-(2.15),

$$\begin{aligned} 2 \times \text{Average '}\frac{1}{2}\text{' = } \frac{(S_7 + \dots + S_1 + S_0)}{4} \\ = \text{Average '0' + Average '1'} \\ \approx \text{Average '1'} \end{aligned} \quad (2.16)$$

which means  $2 \times \text{Average '}\frac{1}{2}\text{'}$  can be used to approximately represent Average '1'.

It can be rewritten as

$$\text{Average '1' = } h_{i,7} \dots h_{i,1} h_{i,0} \approx \frac{(S_7 + \dots + S_1 + S_0)}{4} \quad (2.17)$$

An example is given in Fig. 2.24 to show the amplitudes of  $S_i$  and these signals. The set of  $S_i$  is provided by Matlab. The average values are obtained from Equations (2.12)-(2.14), (2.16). The horizontal axis is the order of  $S_i$ , and the vertical axis shows the decimal representation of the amplitude of  $S_i$ . From Fig. 2.24, the assumption in (2.15) and the deduction in (2.17) are illustrated.

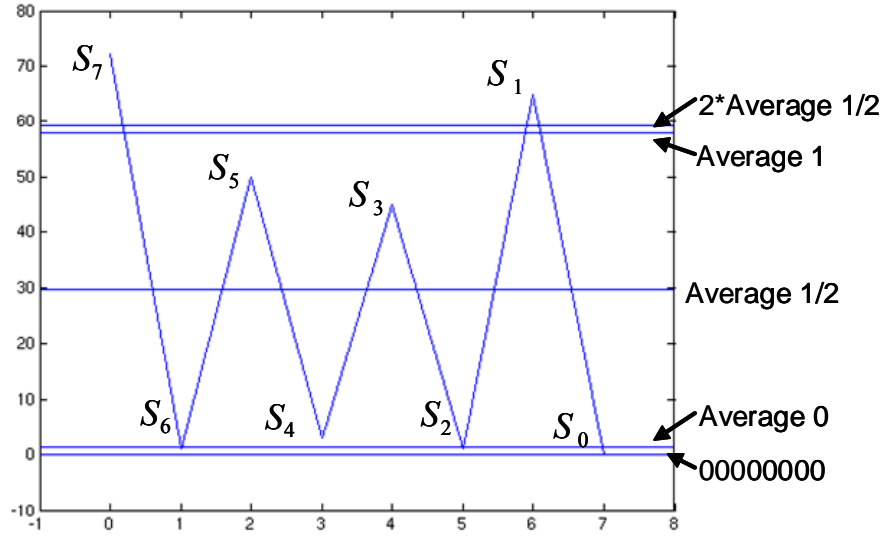


Fig. 2.24 An example of  $S_7 \dots S_1 S_0$  with SNR = 12

Equations (2.15) and (2.17) form the principle of the amplitude estimation. So Equations (2.10) and (2.11) turn out to be

$$\begin{aligned}
 sim_i &= s_{i,7} \dots s_{i,1} s_{i,0} - l_{i,7} \dots l_{i,1} l_{i,0} \quad \text{when } b_i = 1 \\
 &= s_{i,7} \dots s_{i,1} s_{i,0} - \text{Average '0'} \\
 &\approx s_{i,7} \dots s_{i,1} s_{i,0} - "00000000"
 \end{aligned} \tag{2.18}$$

$$\begin{aligned}
 sim_i &= h_{i,7} \dots h_{i,1} h_{i,0} - s_{i,7} \dots s_{i,1} s_{i,0} \quad \text{when } b_i = 0 \\
 &= \text{Average '1'} - s_{i,7} \dots s_{i,1} s_{i,0} \\
 &\approx \frac{(S_7 + \dots + S_1 + S_0)}{4} - s_{i,7} \dots s_{i,1} s_{i,0}
 \end{aligned} \tag{2.19}$$

The result is to be substituted in (2.9). The correlation equation and amplitude estimation is used in the synchronizer and correlator.

In Fig. 2.25 it is the block diagram with the amplitude estimation.



Fig. 2.25 Block diagram of DBB subsystem with amplitude estimation

It is obvious that the channel condition influences the amplitude of  $S_i$ , and further influences the correlation results. It is easy to understand that a static threshold can not satisfy different channel situations to get an always right decision. So we calculate the thresholds on the fly, as

$$p\_threshold = TC \times \text{Average '1'} \tag{2.20}$$

$$c\_threshold = TC \times \text{Average '1'} \quad (2.21)$$

where  $TC$  is the threshold coefficient for the synchronizer and correlator. The value selection of  $TC$  is discussed in next subsection.

## 2.5.4 Simulations on threshold coefficients

In order to find out which value  $TC$  most probably belongs to, we tried different input packets with right and wrong preambles respectively. Notice that the right preamble is “11110000”.

$clk2x$  is the clock signal whose frequency is 200 kHz.  $f\_out3$  is the output of the matched filter, i.e., the input signal of the synchronizer.  $p\_average$  stands for Average ‘1’ in Equation (2.17).  $p\_thres$  is the threshold calculated on the fly via Equation (2.20). The rising edge of  $p\_enable$  indicates the synchronization is done.  $p\_out$  is the output port to transport the rest of the packet to the next block after synchronization. All numbers are expressed in hexadecimal.

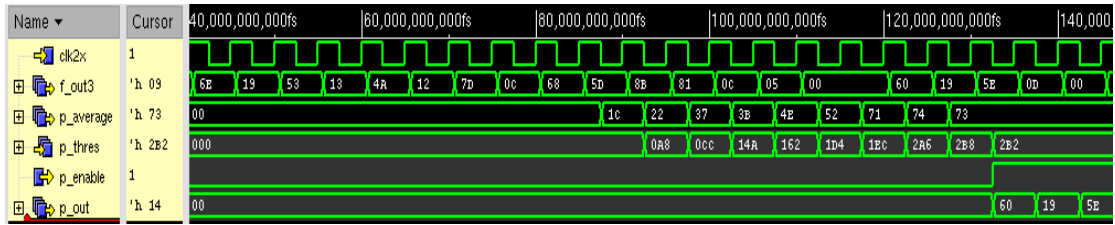


Fig. 2.26 Preamble is “11110000”,  $TC=6$

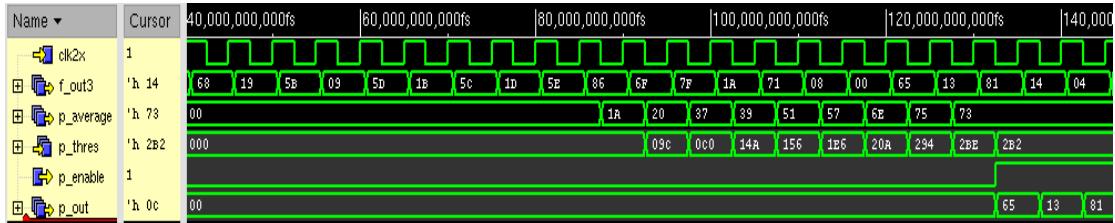


Fig. 2.27 Preamble is “11110100”,  $TC=6$

From Fig. 2.26 and Fig. 2.27, we can see the block is synchronized at a right preamble, as well as a one-bit-wrong preamble when  $TC = 6$ . It indicates that the chosen threshold is too small.

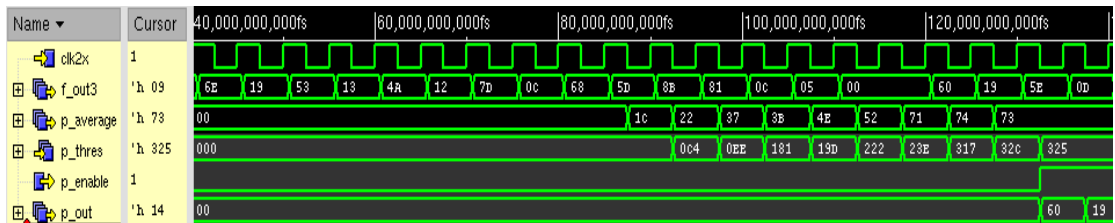


Fig. 2.28 Preamble is “11110000”,  $TC=7$



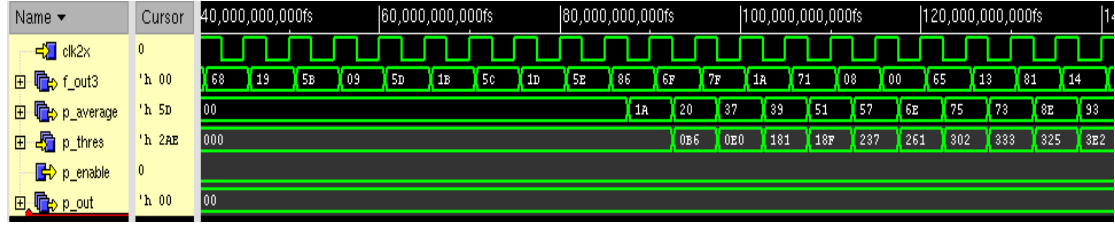


Fig. 2.29 Preamble is “11110100”,  $TC = 7$

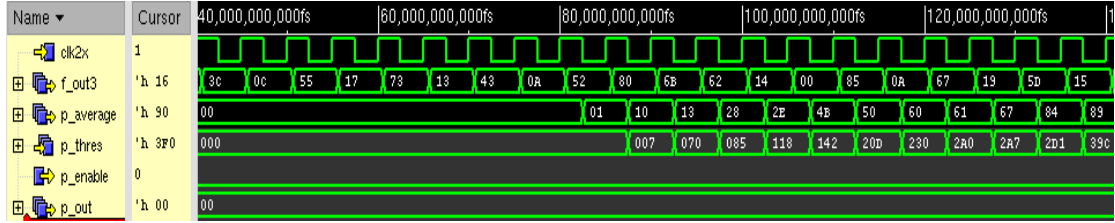


Fig. 2.30 Preamble is “11110010”,  $TC = 7$

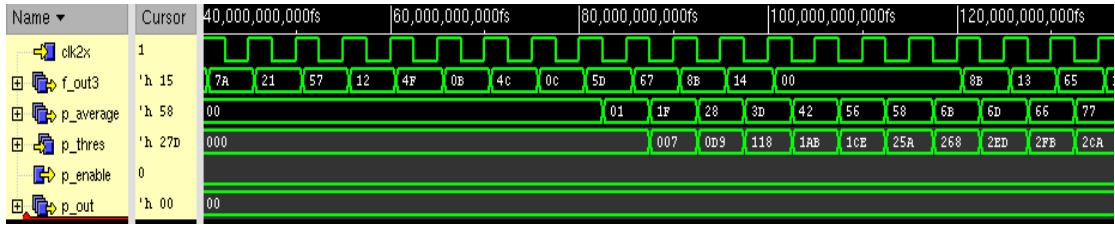


Fig. 2.31 Preamble is “11100000”,  $TC = 7$

Judging from Fig. 2.28-Fig. 2.31,  $TC = 7$  is a suitable value since the synchronization only happens when the correct preamble is provided. All the possible one-bit-wrong preambles are verified. However, for the sake of brevity, not all simulation results are provided here. Nevertheless, it is easy to understand that, for any preambles with more than one wrong bit, the block won't be synchronized.

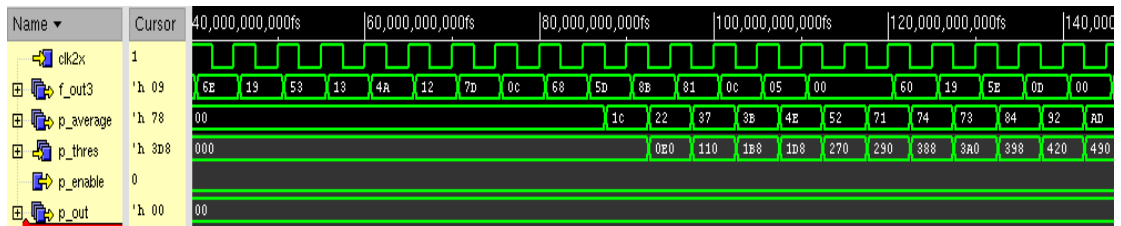


Fig. 2.32 Preamble is “11110000”,  $TC = 8$

For  $TC = 8$ , the detection is missed when the correct preamble is sent. Hence it is not a suitable value for  $TC$ . Eventually  $TC = 7$  is chosen.

$$p\_threshold = 7 \times \text{Average '1'} \quad (2.22)$$

For the same reason, we choose

$$c\_threshold = 7 \times \text{Average '1'} \quad (2.23)$$

## 2.6 Summary

In this chapter, the fundamental implementation and blocks are discussed. There are four main blocks, the matched filter, synchronizer, Manchester decoder and correlator. They work together to optimize, synchronize and decode the received packet, and finally obtain a trigger to the main radio when the packet is targeted to this sensor node. The clock frequency is divided step by step in the chain to reduce the power consumption. The necessity of the amplitude estimation is discussed. The equations of the softbit correlation and threshold coefficient have been deduced. The linear relationship between the thresholds and amplitude estimation is determined by trial and error.

## Other implementation details

---

The previous chapter provides an overview of implementation. In this chapter, some details are explained, together with design choices which improve robustness and power efficiency of the subsystem.

SPI is introduced for testing purpose in Section 3.1. In the matched filter, canonical signed digit (CSD) is utilized to reduce the non-zero digit in the tap coefficients. Section 3.2 explains why and how to apply the CSD algorithm in the coefficient computation. Section 3.3 indicates why the filtered data needs to be downsampled and split into four groups. In Section 3.4, the application of clock gating is analyzed. Section 3.5 explains how multiple threshold voltage libraries optimize the leakage power. In Section 3.6, the link information extraction is explained. The whole chapter is summarized in Section 3.7.

### 3.1 Cooperation with SPI

The Serial Peripheral Interface (SPI) Bus is an industry-standard digital interface to write to or read from the internal registers in the design. The detailed description of the SPI circuit we used can be found in the IMEC TN [28].

In order to increase the flexibility and shorten the implementation period, some parameters need to be read out and overwritten during the testing to obtain the best performance. So the SPI is used as the supplementary equipment in measurement.

A simplified SPI block diagram is shown in Fig. 3.1, with external I/O pins and the internal interface. The major function of the SPI in the digital baseband subsystem is to write to the internal registers Po\_Reg1-5 and to read from the internal registers Pi\_Reg16-19 as shown in Table 3.1. The internal registers Po\_Reg1-5 and Pi\_Reg16-19 are used to connect to the internal register in the digital baseband subsystem through internal connections shown in Table 3.1.

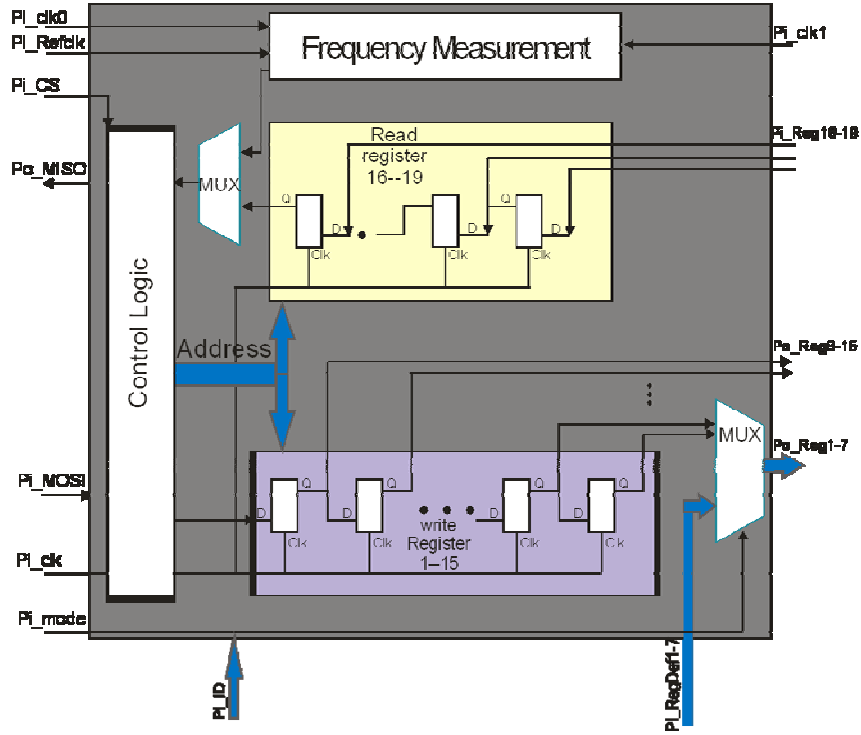


Fig. 3.1 Block diagram of SPI [29]

Table 3.1 Connections between SPI and DBB

SPI	DBB (mode =1)	
Po_Reg1	address	
Po_Reg2	p_threshold (high)	
Po_Reg3	p_threshold (low)	
Po_Reg4	c_threshold (high)	
Po_Reg5	c_threshold (low)	
Po_Reg6	control bits (for testing)	
SPI	DBB	
	mode=0	mode=1
Pi_Reg16	Average '1' 1	link_info (high)
Pi_Reg17	Average '1' 2	link_info (low)
Pi_Reg18	Average '1' 3	
Pi_Reg19	Average '1' 4	

Po\_Reg6 (7 downto 0) is used for control bits describing the communication details between SPI and DBB. The predefined values of Po\_Reg6 are shown in Table 3.2.

Table 3.2 Definitions of control bits written in Po\_Reg6

Po_Reg6(0)=0	load local address to DBB block
Po_Reg6(0)=1	overwrite new address via SPI
Po_Reg6(1)=0	thresholds are determined by Equation (2.22) and (2.23)
Po_Reg6(1)=1	overwrite the thresholds via SPI
Po_Reg6(2)=0	read amplitude estimation via SPI
Po_Reg6(2)=1	read link information via SPI

There is an external pin *mode*. If *mode* =1, local *address* and thresholds (*p\_threshold* and *c\_threshold*) can be set by SPI separately, otherwise the thresholds are set by

Equation (2.22) and (2.23), and the default local address which is randomly chosen as “00101110” is applied. The read out results of *Average ‘1’* or the *link information* (see Section 4.6) are stored in Pi\_Reg16-19.

In conclusion, the top level cell diagram of the digital baseband subsystem is shown in Fig. 3.2. The pins displaying in *italic* on the top and right side are internal connections with SPI.

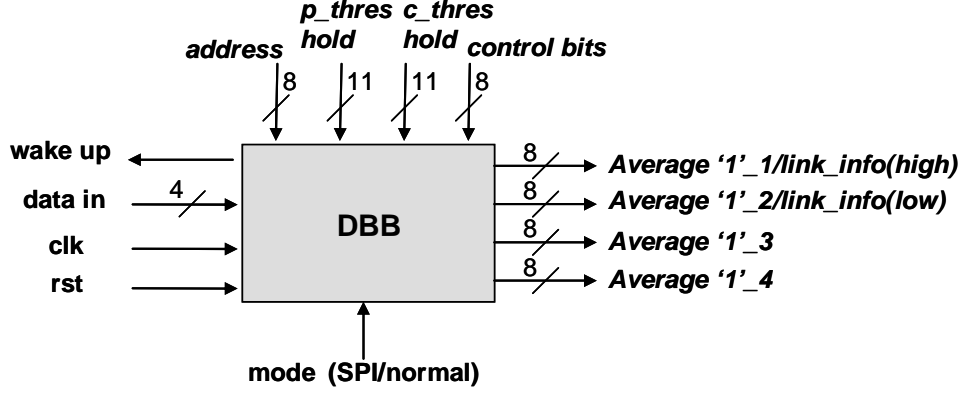


Fig. 3.2 Top level cell diagram of DBB subsystem

### 3.2 CSD coefficients in filter

Canonical signed digit (CSD) number representation indicates a method to describe signed digit numbers without containing adjacent nonzero digits. We will show an algorithm that can minimize the number of nonzero digits compared to the 2’s complement representation for the same value. In this way the complexity of the hardware implementation of a FIR digital filter is reduced since less nonzero digits require less addition operations. In [30] it is proven that multiplication of a single  $k$ -bit multiplicand by  $n$   $k$ -bit multipliers can be performed using  $0.306nk$  additions for CSD numbers, while the binary case requires  $0.375nk$  additions. The power consumption is reduced by using CSD representation as the filter tap coefficients.

To encode 2’s complement number into CSD number, a digit set  $\{\bar{1}, 0, 1\}$  is used.  $\bar{1}$  stands for value ‘-1’. For a 3-bit number, 3 is equal to  $10\bar{1}$  and -2 is equal to  $0\bar{1}0$  in CSD as opposed to their 011 and 110 2’s complement representation. Since the adjacent CSD digits are never both nonzero, there are at most  $\left\lfloor \frac{n}{2} \right\rfloor$  nonzero digits for an  $n$ -bit number [31].

CSD encoding is to analyze pairs of adjacent digits of the 2’s complement number from the LSB (least significant bit)  $x_0$  to the extended MSB (most significant bit)  $x_n^*$ . According to [31] if the number is negative,  $x_n^*$  is 1, otherwise it is 0. Table 3.3 and Fig. 3.3 show the conversion rules from 2’s complement number  $X$  to CSD number  $C$ , where  $X = x_n^*x_{n-1}x_{n-2}x_{n-3} \dots x_3x_2x_1x_0$  and  $C = c_{n-1}c_{n-2}c_{n-3} \dots c_3c_2c_1c_0$ .

Table 3.3 Look-up table of CSD conversion [31]

carry-in	$x_{i+1}$	$x_i$	carry-out	$c_i$
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	1	$\overline{1}$
1	0	0	0	1
1	0	1	1	0
1	1	0	1	$\overline{1}$
1	1	1	1	0

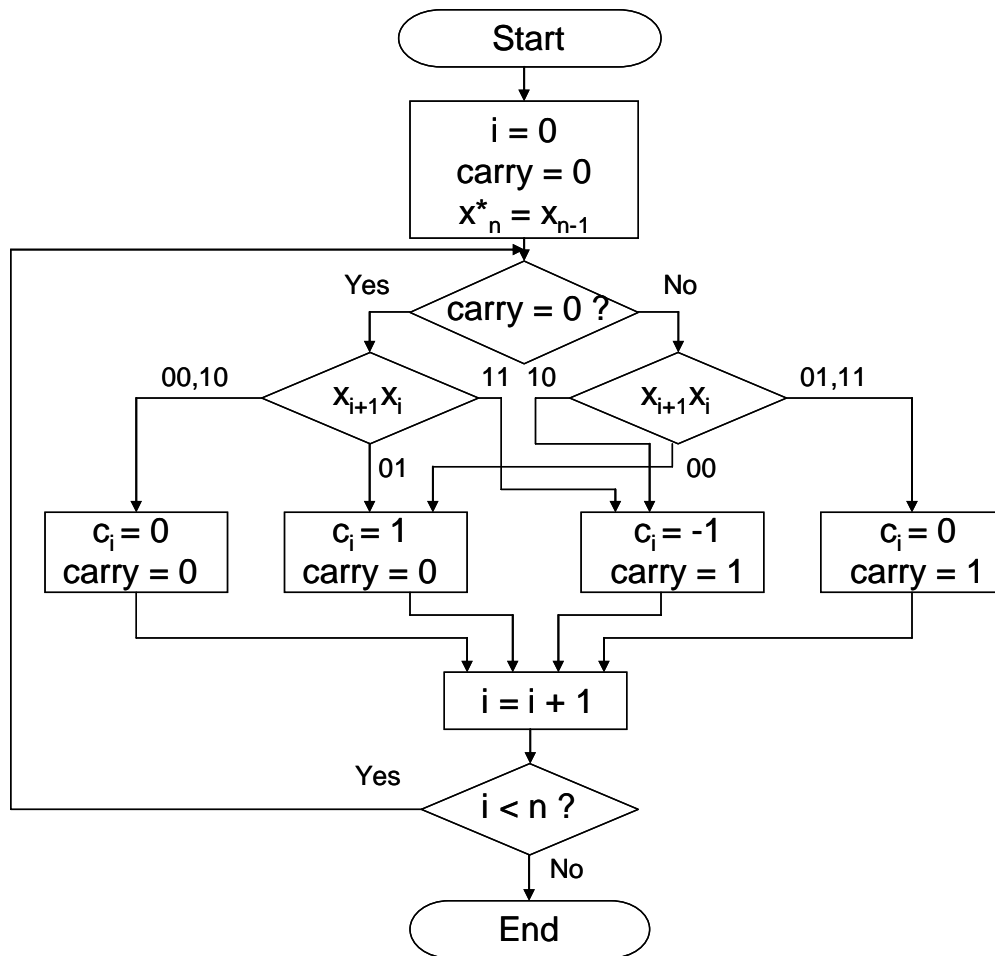


Fig. 3.3 Conversion flow of 2's complement to CSD [31]

Matlab has been used to do the conversion. Resulting tap coefficients of the matched filter are shown below in Table 3.4. As discussed in the previous chapter the matched filter originally had 25 taps, some of which came out to be relatively small and can be ignored. As a consequence the number of taps is reduced to 5. Notice that the coefficients are symmetric.

Table 3.4 Conversion with coefficients

	tap 1	tap 2	tap 3	tap 4	tap 5
decimal	0.2122	0.5	0.6366	0.5	0.2122
CSD	010101	100000	101000	100000	010101

In order to evaluate the CSD coefficients, we compared it with a set of reference coefficients, which represents a high accuracy filter with the order of 32. The result of cross-correlation is depicted in Fig. 3.4. The horizontal axis presents time normalized to symbol period.

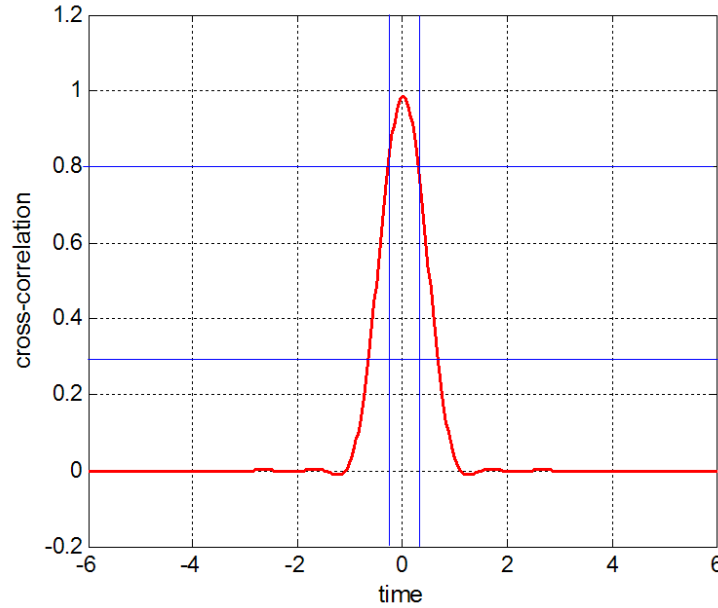


Fig. 3.4 Comparison of CSD and high accuracy coefficients

According to the IEEE 802.15.4a standard [32], the requirement for the coefficients is that it should have “a magnitude of the cross-correlation function whose main lobe is greater or equal to 0.8 for a duration of at least  $T_0 / 4$ , and any side lobe shall not be larger than 0.3”. This criterion is also suitable for narrowband communication [33]. Fig. 3.4 shows that the CSD coefficients meet this requirement.

### 3.3 Quadruple signal processing

In Section 2.1 and 2.2, it is mentioned that the data is four times downsampled in the matched filter. In the realization the data is split into four groups, instead of sampling only one output out of four. The reason of doing this is explained as follows.

Fig. 3.5 shows the Matlab simulated output of the matched filter before downsampling. The original signals are “11110000 10101010 10101001 10011001”. Each symbol is sampled 4 times so that the output contains four times as many as signals. The vertical axis describes the decimal amplitude of the 8-binary-bit output. The horizontal axis labels the outputs in the chronological order.

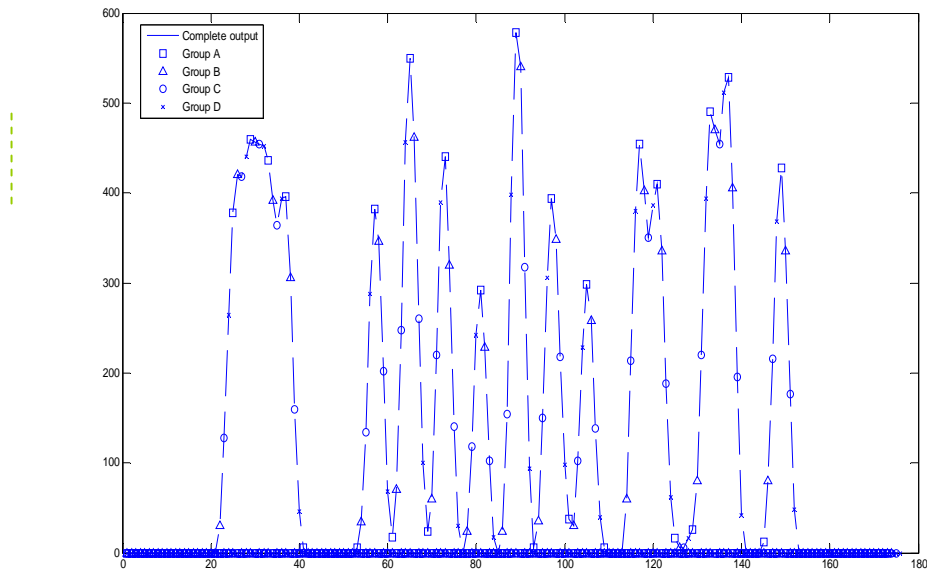


Fig. 3.5 Output data of filter with four different colors indicating four groups

It can be seen from the figure above that the filter outputs are divided into four groups with different markers. Any group can be seen as a set of four times downsampled result. The Group A describes the sequence “11110000 10101010 10101001 10011001” with the best performance, while the Group C has the poorest performance of recovering the correct data.

The problem at present is how to select the data of Group A and avoid sending the Group C to the next block. However, it is not possible to distinguish the Group A from the whole filtering output. Our solution is to keep all the four groups and process them separately.

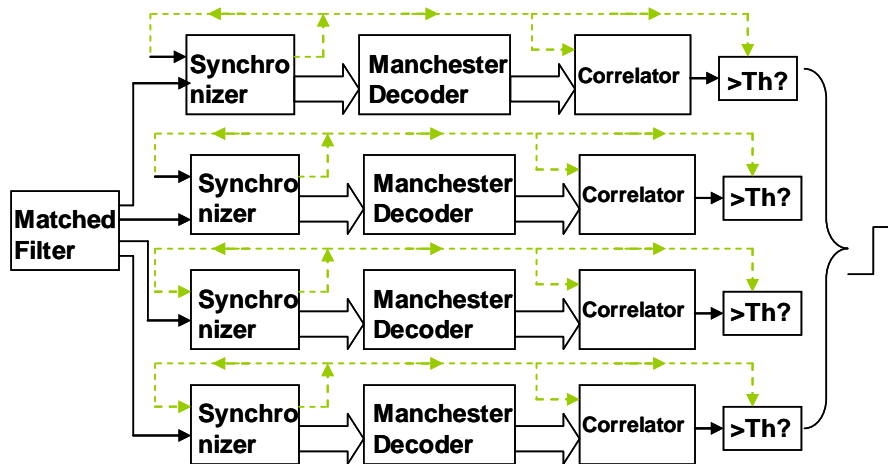


Fig. 3.6 Block diagram of DBB subsystem splitting into four groups

The data of Group A should have the biggest correlation result if the beacon packet is targeted to this sensor node. We assume the trigger signal is produced when any one



of the four correlation results is larger than the threshold. Therefore we won't miss the Group A.

The implementation of the four signal processing chains increases the robustness significantly, and the power consumption is still within the budget.

### 3.4 Clock gating

Clock gating is one of the conventional energy saving methodologies used in synchronous circuits on the synthesis level. The clock results in two major parts of power consumption [34]: (1) power consumed by flip-flops (FFs); and (2) power consumed by the clock buffer tree. It is a good idea to partially suspend the clock when there is no need to change the output. When the clock is disabled, the dynamic power becomes zero and only leakage current exists, while the original functionality of the circuit is still maintained.

It is a two-step process for RTL clock gating [35]. Firstly, the enable terms are identified. The clocks to all the FFs and logic circuit sharing the same enable terms will be gated. The second step is to insert the clock gating cells into the clock tree using the enable logic. It is accomplished by RTL synthesis tools. There are two possible ways to implement clock gating [34]: the latch-free clock gating and the latch-based clock gating.

The latch-free clock gating uses simply a single AND or OR gate. Here we take the AND gate as an example, when the enable signal (EN) is '0', the gated result is always '0'. If EN='1', the gate appears transparent to the clock signal.

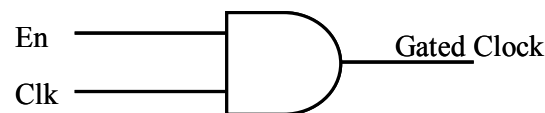


Fig. 3.7 Latch-free clock gating

Some glitches will be produced if the transition is asynchronous. So the latch-free clock gating is inappropriate for single-clock flip-flop based design.

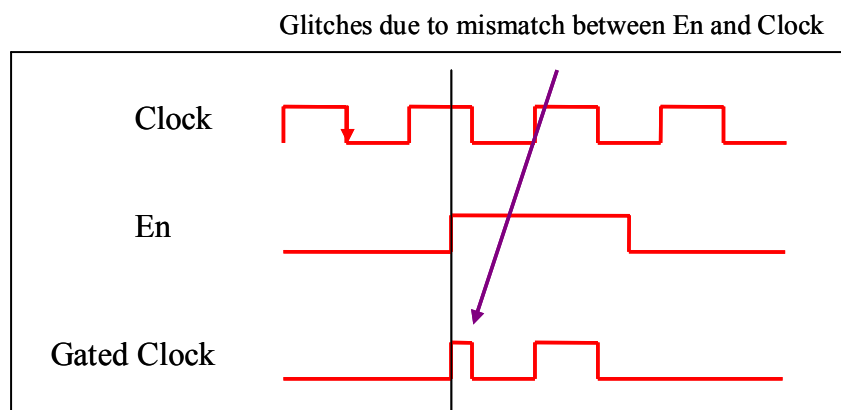


Fig. 3.8 Glitches due to mismatch between En and Clock

A level-sensitive latch is added to hold the enable signal from the rising edge of the clock. Hence the output signal is synchronized with the clock, and the glitches are avoided in the gated clock.

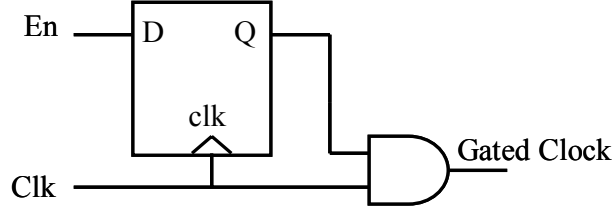


Fig. 3.9 Latch-based clock gating

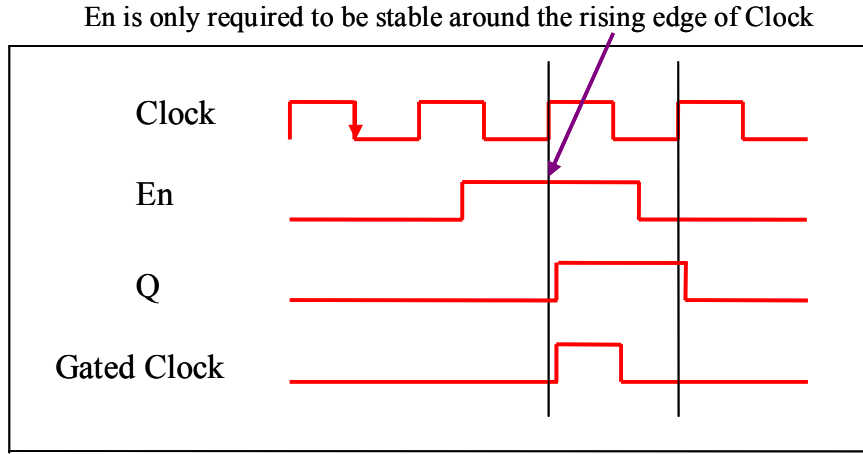


Fig. 3.10 The glitches are eliminated by the latch

There is little designer involvement in the RTL level required for the clock gating. In synthesis phase the certain clock gating cells will be inserted automatically by EDA tools. Due to the cascade structure and discontinuous operation schedule in the subsystem, benefit can be obtained from the clock gating. Moreover, the area and timing constraints are relaxed in our design. Consequently, the impact on area and timing can be ignored and the latch-based clock gating is applied.

### 3.5 Multiple $V_T$ optimization

Multiple threshold voltage libraries are used to optimize the leakage power without compromising the timing performance.

$$P_{leakage} \propto V_{DD} e^{-V_T / nV_{tm}} \quad (3.1)$$

where  $V_{DD}$  is the supply voltage,  $V_T$  stands for the threshold voltage,  $n$  is the subthreshold swing coefficient, thermal voltage  $V_{tm} = kT / q$ ,  $k$  is Boltzmann constant,  $T$  is absolute temperature,  $q$  is the magnitude of the electrical charge on the electron [36]. As we can see from Equation (3.1), the larger the  $V_T$  is, the smaller the leakage.

There are three types of transistors with different thresholds available in our library: regular  $V_T$ , high  $V_T$  and low  $V_T$ . The high threshold transistor is slower but leaks less, and can be used in non-critical circuits to reduce the static power. And the low

threshold standard cells, which consist of transistors with high speed and relatively large leak current, are used in critical paths. Since the priority of this design is power saving and the timing requirement is relaxed, the high  $V_T$  is chosen. In addition, due to the constraint of the digital design flow, the regular  $V_T$  is also chosen.

### 3.6 Link information extraction

After the wake up trigger is produced, the system can be used for extracting other useful information from beacon packet as well. For instance, the information on channel selection or device selection can be carried into the packet. This kind of information helps to configure the main radio properly at the beginning stage after it is woken up.

For this purpose, the packet has to be extended as follows. Firstly, a flag bit is added, in which '1' indicates that a valid link information sequence is contained. If the flag bit is '0', no additional information is included. After the flag bit of '1', we assume a 10-symbol sequence describing some link parameters is given. Together with the wakeup address, this extra sequence is Manchester encoded.

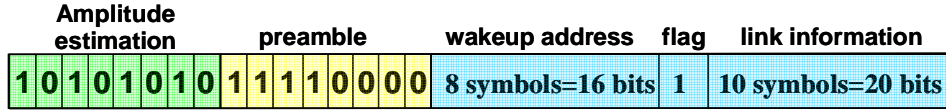


Fig. 3.11 Beacon packet with additional flag and link information

At the transmitter end, the link information (such as channel and device selection) is loaded behind the wakeup address sequence in the beacon packet. At the receiver end, if the wakeup trigger is produced and the flag bit is '1', then the link information will be shifted to a hard decision block to convert the soft bit information  $S_{link,i}$  into hard bits  $H_{link,i}$ . Half of Average '1' is employed during hard decision as in Fig. 3.12.

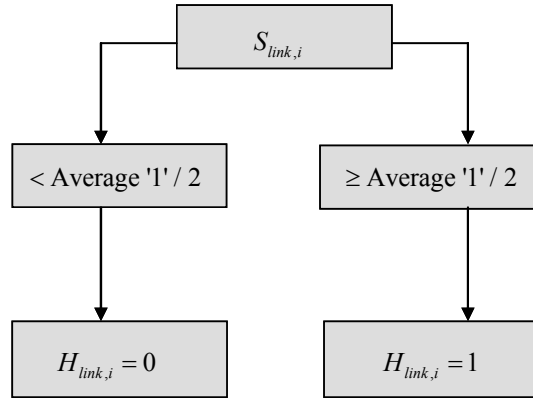


Fig. 3.12 Behavior description of hard decision block

After that, the link information in hard bits is provided to external components, such as MCU, as shown in Fig. 3.13.

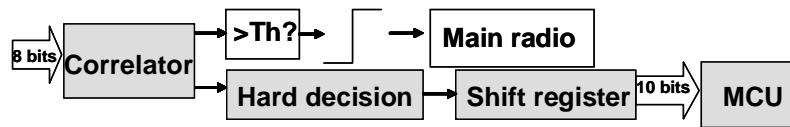


Fig. 3.13 Processing flow of link information

### 3.7 Summary

This chapter explains several strategies in order to make the flow provided in chapter 2 work effectively with ultra-low dynamic and leakage power. The peripheral equipment is also shown.

## Simulation results and analysis

---

In the previous two chapters, the basic implementations and details of the digital baseband subsystem have been explained. The subsystem is implemented in VHDL via the “RTL to GDS2 flow” [37] at IMEC-NL. Owing to the time constraints during design period, there are two versions of implementations. In the first version (Ver1),  $p\_threshold$  and  $c\_threshold$  are constant. It is regarded as an intermediate version without the amplitude estimation. In the second version (Ver2), the thresholds follow the relationship in Equations (2.22) and (2.23).

The input data is generated by the Matlab script with the parameters shown in Table 4.1.

Table 4.1 Parameters for simulating transmitter and channel in Matlab

Transmitter	sampling rate $nsamp = 4$
	roll-off factor $\alpha = 1$
Channel	Additive White Gaussian Noise, SNR = 12

The simulation results are presented and analyzed in this chapter. In the measurement our design is co-working with SPI to increase flexibility, and the testing flow is shown in Section 4.1. Section 4.2 and 4.3 provide some simulation data we obtained. Due to the low data rate, the timing target is relaxed, so there is no timing issue in our design. The chapter is concluded in Section 4.4.

### 4.1 Testing flow

During the cooperation with SPI, the performance of the subsystem can be measured based on several address or thresholds conditions. The testing procedure in the cooperation between SPI and DBB subsystem is provided as follows.

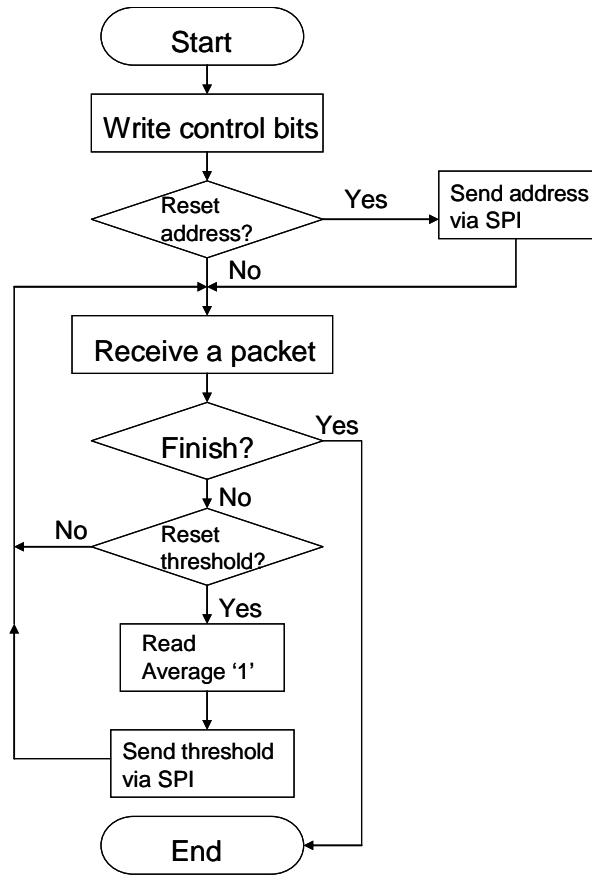


Fig. 4.1 Testing flow with SPI

## 4.2 Synthesis results

The number of gates in our design after the synthesis by RTL-compiler from Cadence is shown in Table 4.2.

Table 4.2 Number of gates after synthesis

Design Version	Sequential	Inverter	Buffer	Logic	Total Num.
Ver1	930	363	33	2223	3549
Ver2(no SPI)	1114	434	33	3684	5265
Ver2(with SPI)	1379	473	2	4207	6061

## 4.3 Power consumption

The results of post-layout power simulation (worst case) using Cadence ncsim and Synopsys PrimeTime are shown in Table 4.3.

Table 4.3 Post-layout power consumption (1)

		Switching( $\mu$ w)	Internal( $\mu$ w)	Dynamic( $\mu$ w)	Leakage( $\mu$ w)	Total( $\mu$ w)
Ver1	①	1.39	3.34	4.73	1.66	6.39
	②	1.4	3.5	4.9	1.67	6.57
	③	1.19	3.19	4.38	1.66	6.04
Ver2 without SPI	①	2.47	6.38	8.85	2.31	11.16
	②	2.38	6.29	8.67	2.32	10.99
	③	2.23	6.13	8.36	2.32	10.68
Ver2 with SPI	①	2.66	6.26	8.92	2.84	11.76
	②	2.43	6.08	8.51	2.83	11.34
	③	2.18	5.89	8.07	2.83	10.9

Simulation condition ①: the circuit is not synchronized (i.e. the decoder and correlator are inactive). It shows the power consumption of the subsystem during a wrong preamble.

Simulation condition ②: the simulation time is limited in only one packet period (approx. 200  $\mu$ s). It describes the active power of our design (i.e. all blocks are active).

Simulation condition ③: It shows the average power spreading over one packet period (approx. 200  $\mu$ s) plus a short idle period (100  $\mu$ s).

We can see from Table 4.2 and Table 4.3 that the amplitude estimation and SPI add complexity and power consumption to the design. The power consumptions under simulation conditions ① and ② in both Ver1 and Ver2 are close to each other, which mean energy consumption doesn't vary much from unsynchronized state to synchronized state. Moreover, as an average measurement value, power consumption is slightly less when a longer idle period is applied when we compare results under conditions ② and ③. So it is shown that the length of idle period is a trivial factor in power. Table 4.3 also indicates that SPI only adds less than 1  $\mu$ w power to the whole design.

Table 4.4 Post-layout power consumption (2)

	Dynamic( $\mu$ w)	Leakage( $\mu$ w)	Total( $\mu$ w)
Ver2 without clock gating	8.73	2.85	11.58
Ver2 with clock gating	8.07	2.83	10.9

Table 4.4 shows the power simulation results with and without clock gating. It proves that the dynamic power is reduced dramatically due to clock gating.

## 4.4 Summary

In this chapter, the simulation flow is provided and the simulation results are analyzed. Due to the low data rate, the timing target is easily reached, so there is no timing issue.





# Hardware implementations

---

We have proposed two ways to implement the hardware of the digital baseband subsystem: field-programmable gate array (FPGA) and application-specific integrated circuit (ASIC). In Section 5.1, the FPGA realization is investigated. The design was fabricated in TSMC90 technology in February and June 2009 at IMEC-NL. The measurement setup and results of the tape-out are shown in Section 5.2. The chapter is concluded in Section 5.3.

## 5.1 FPGA verification

FPGAs are highly flexible logic devices, which are programmable by configuration files from particular design environment. We used Xilinx XC3S200 Spartan-3 FPGA board for a quick verification of our design. Although FPGAs are not optimized for certain functionality, they avoid high initial cost, lengthy development cycles, and inherent inflexibility of conventional ASICs. In addition, the programmability of FPGA permits design upgrades in the field without hardware replacement [38]. The design tool installed in PC is Xilinx integration software environment (ISE) 6.2i.

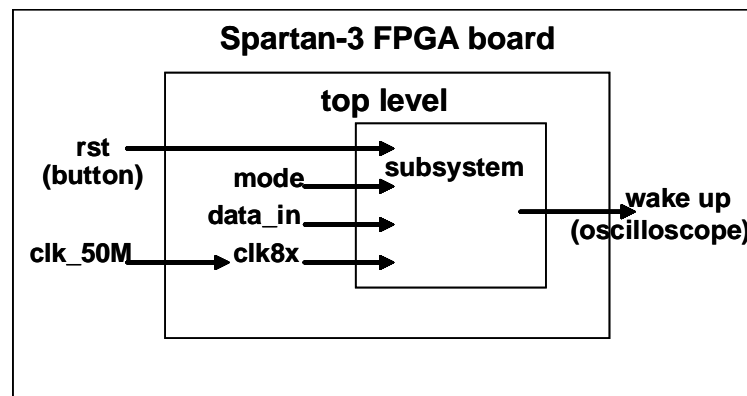


Fig. 5.1 Scheme of FPGA implementation

For simplicity, the subsystem on FPGA was realized without SPI. The top level file combines our subsystem and test data as input. The internal clock provided by the oscillator has a dedicated frequency 50 MHz. It is divided into the required 800 kHz clock. The reset pin (high active) is assigned to a button on the board, which is high voltage during pressing. The output signal is connected to an oscilloscope, whose waveform is shown in Fig. 5.. It is a trigger signal ( $0V \rightarrow 3.3V$ ) corresponding to a beacon packet targeting at the node with the local address “00101110”.

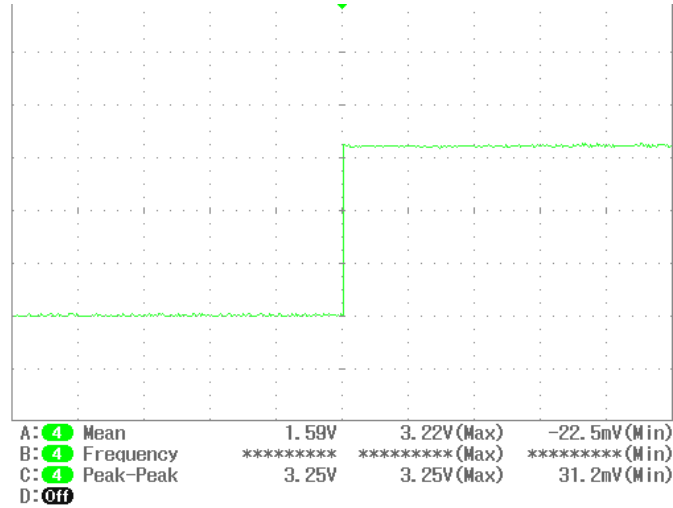


Fig. 5.2 Wakeup trigger from subsystem on Spartan-3 FPGA

Table 5.1 presents the device utilization abstracted from the synthesis report.

Table 5.1 Device utilization summary

Number of external IOBs	8 out of 173	4%
Number of LOCed external IOBs	3 out of 8	37%
Number of RAMB16s	1 out of 12	8%
Number of slices	1342 out of 1920	69%
Number of BUFGMUXs	1 out of 8	12%

## 5.2 Tape out measurement

Due to the time constraints during the design period, the subsystem was realized in two steps. In the first version,  $p\_threshold$  and  $c\_threshold$  are constant. In the second version, the thresholds follow the relationship in Equations (2.22) and (2.23). The design was taped out twice in February and June 2009 in TSMC90LP technology.

The February chip, which contains the first version of the digital baseband subsystem, was obtained in June. It combines the DBB part, SPI, ring oscillator together with the RF circuitry. [39] is the technical note describing the chip layout of the February chip, which is given in Fig. 5.3. The die size is  $1850 \times 1850 \mu\text{m}^2$ , including bond pads.

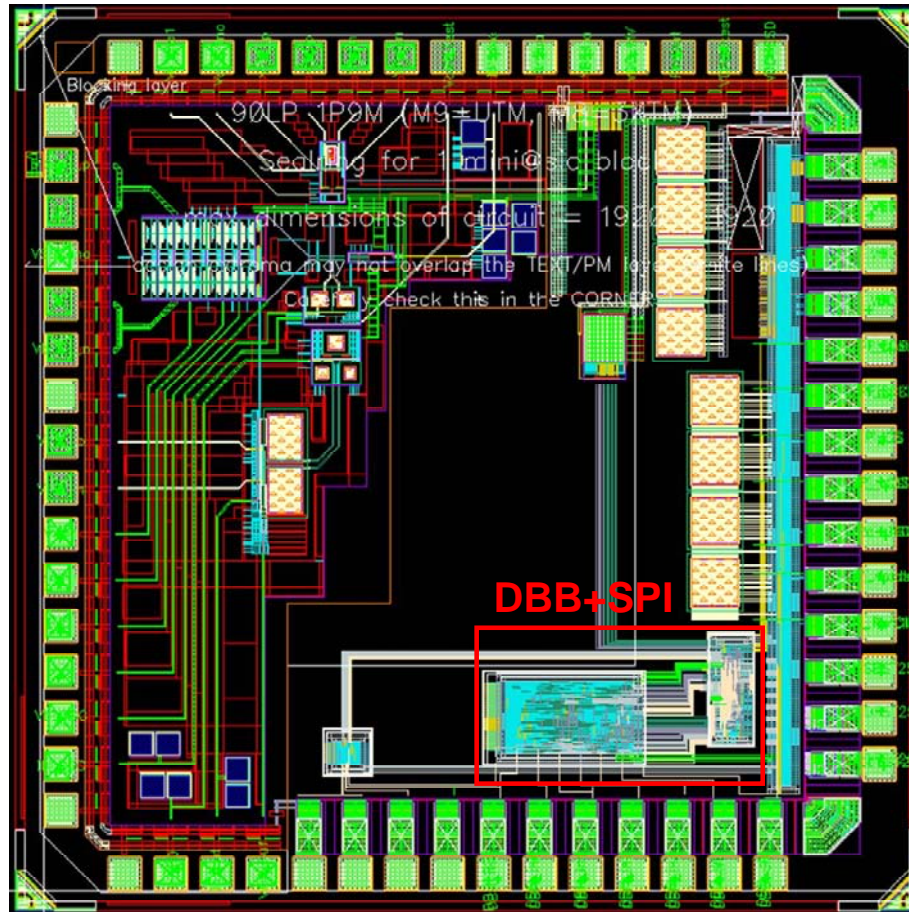


Fig. 5.3 Layout view of Feb tape out [39]

The June chip contains the second version based on the first one. [40] is the technical note describing the chip layout of the June chip, as shown in Fig. 5. DBB and SPI are placed on an individual power ring. The area figures of this design are presented in Table 5.2.

Table 5.2 Area of June design

standard-logic area (without power rings)	$250 \times 250 \mu\text{m}^2$
standard-logic area (with power rings)	$325 \times 325 \mu\text{m}^2$
total area with IO-ring (without bond pads)	$590 \times 790 \mu\text{m}^2$

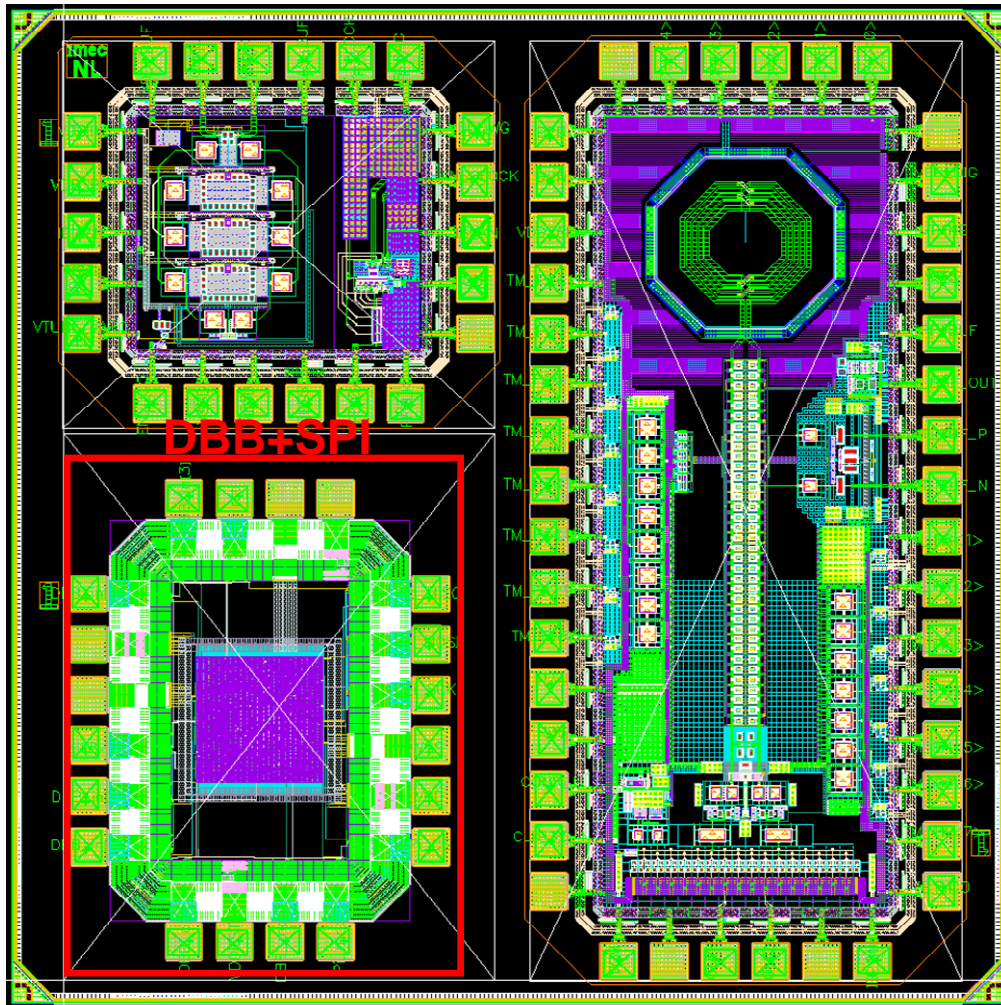


Fig. 5.4 Layout view of June tape out, DBB subsystem is in the lower left ring [40]

In order to measure the power consumption of the chips, the measurement setup is demonstrated in Fig. 5. and Fig. 5..

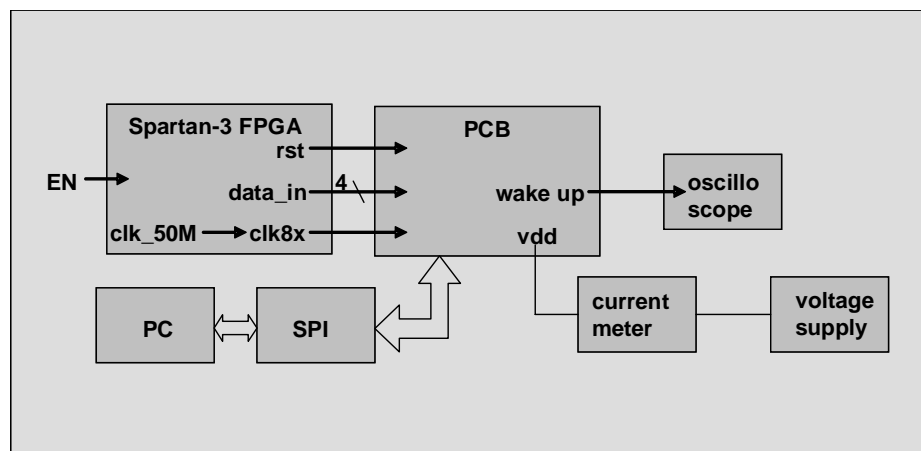


Fig. 5.5 Scheme of measurement system

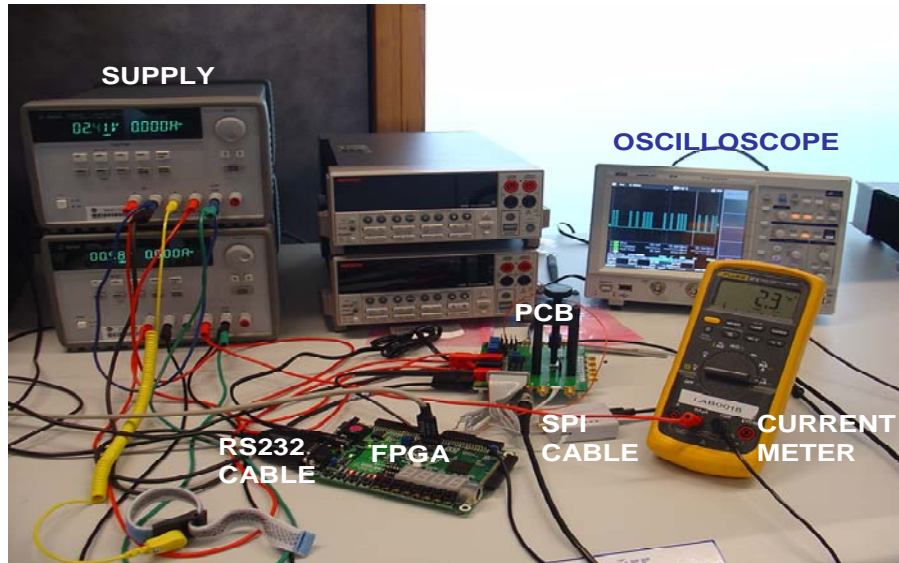


Fig. 5.6 Measurement system in the lab

The FPGA provides the desired inputs including the data, clock and reset signal to the chip which is clamped to the PCB. These inputs are controlled by an enable button (low voltage active) on the FPGA. If the button is pressed to obtain a high voltage, all inputs remain zero, and only the leakage current exists in the circuit. There are ten different wakeup packets stored in the FPGA, which are sent to the chip repeatedly. The idle time between packets is approximately 80 clock cycles (100  $\mu$ s), after which the circuit is reset.

The FPGA sends ten packets with different addresses to the subsystem, whose order is shown in Table 5.3.

Table 5.3 Input and expected output in the measurement

Input	R	R	R	W	W	W	R	W	R	W
Expected output	1	1	1	0	0	0	1	0	1	0

In the table above, “R” stands for a right packet with exactly the same address as in the chip, “W” stands for a wrong packet with a different address. According to this input order, we captured the wakeup results on the screen of an oscilloscope as Fig. 5.7. The result sequence appears in the order as 11100010101110001010....Therefore the functionality of the design is proven.



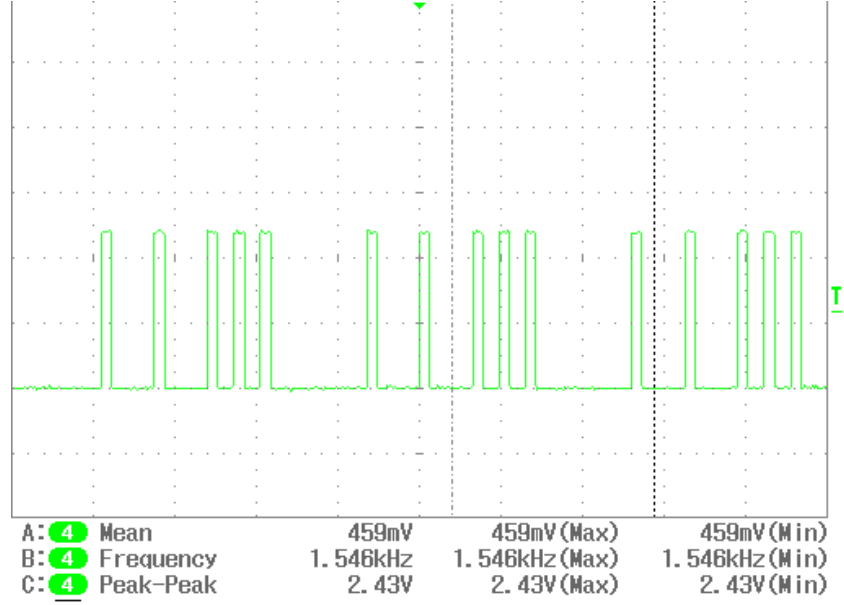


Fig. 5.7 Wakeup triggers from subsystem on chip

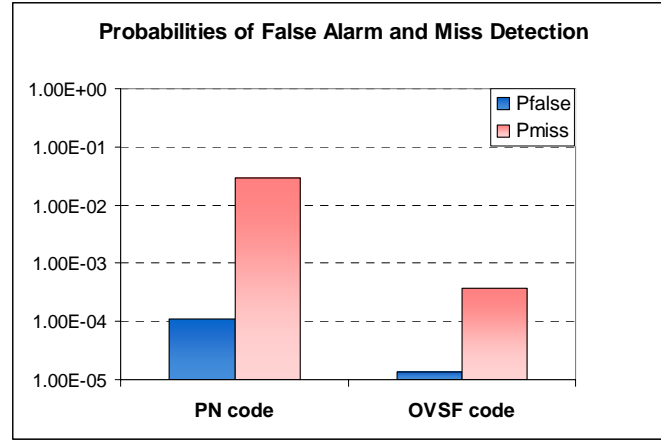


Fig. 5.8 Probabilities of false alarm and miss detection

Fig. 5.8 shows the probabilities of false alarm and miss detection in the June chip with different address codes.

Next, we measured the current flowing through the digital circuit and calculated the power consumption. The data of the June chip is recorded in Table 5.4.

Table 5.4 The measurement values: current and power

$V_{DD}$ (V)	$I_{dyn}$ ( $\mu$ A)	$I_{leak}$ ( $\mu$ A)	$I_{total}$ ( $\mu$ A)	$P_{dyn}$ ( $\mu$ W)	$P_{leak}$ ( $\mu$ W)	$P_{total}$ ( $\mu$ W)
1.2	6.5	0.5	7	7.8	0.6	8.4
0.9	4.7	0.35	5.05	4.23	0.315	4.545
0.6	3.3	0.2	3.5	1.98	0.12	2.1

When we compare the power consumed in the circuit, it is obvious that lower supply voltage leads to lower power, which follows the equation from [41]

$$P_{dyn} = C_L V_{DD}^2 f_{0 \rightarrow 1} \quad (5.1)$$

where  $C_L$  is the load capacitance  $V_{DD}$  is the supply voltage and  $f_{0 \rightarrow 1}$  represents the frequency of energy-consuming transitions.  $P_{dyn}$  is reduced approximately four times due to half of original  $V_{DD}$ .

If we slightly reduce the supply of the I/O ring from 2.5 V to 2.4 V,  $V_{DD}$  can be even lower. The lowest acceptable  $V_{DD}$  is 0.55 V. However, to maintain the system reliability,  $V_{DD}=0.9$  V is a suitable supply.

### 5.3 Summary

The subsystem is verified by Spartan-3 FPGA and TSMC90 chip. The design is functional as designed. The power consumption meets our design target and follows the equation in digital circuit. The minimal active power achieved is 2.1  $\mu$ w at the data rate 200 kbps.





## Conclusions and future work

---

### 6.1 Conclusions

In this thesis we present the digital baseband subsystem in the wakeup radio, which is used to reduce power consumption for data communication within WSN. The main radio in a sensor node is relatively power hungry, while the battery equipped is limited. Therefore the main radio should be suspended when there is no data communications. Based on this consideration, the idea of the ultra-low-power wakeup radio is proposed. The wakeup radio listens to the channel and releases a wakeup signal to the main radio as soon as a new communication request is detected. The main radio will switch from the sleep mode to the active mode after the wakeup signal is present and fall into the sleep mode again after the communication is terminated. In this way, power consumption is minimized, and a longer lifetime of the sensor node can be achieved.

The wakeup receiver is built to deal with a short packet, which basically consists of a preamble and a wakeup address. Each receiver is assigned to a unique local address. In this way, it can be guaranteed that one packet being transmitted will only wake up one node at a time.

The wakeup radio is composed of three parts, the RF frontend, ADC and the digital baseband. In the scope of this thesis, we focus on the digital baseband. The features of the baseband subsystem include low data rate as well as always-on status. Hence it is required to design and implement a subsystem with simplicity, reliability and high power efficiency.

In Chapter 2, the architecture overview of this design is presented. The cascade processing blocks, including the matched filter, synchronizer, decoder and correlator, are discussed separately. The four blocks work together to optimize, synchronize and decode the received packet, and finally obtain a trigger to the main radio when the packet is targeted to this sensor node. The clock frequency is divided step by step along the processing flow to reduce the power consumption. We also analyze the design requirements and explain the reasons of implementation choices. The necessity of the amplitude estimation is discussed and the implementation details of calculation on the fly are given. The equations of the softbit correlation and threshold coefficient have been deduced. The linear relationship between the thresholds and amplitude estimation is determined by trial and error.

In Chapter 3, the details of each block are explained. Some other power-efficient strategies such as CSD coefficients, clock gating, multiple  $V_T$  libraries optimization, are explained as well. Matlab simulation indicates that the filtered data needs to be

downsampled and split into four groups to enhance the reliability. Besides, some peripheral functions, i.e., the link information extraction and SPI, are discussed.

The subsystem is implemented in VHDL, and the simulation results are presented and analyzed in Chapter 4. The simulation flow is provided and the results are analyzed. Due to the low data rate, the timing target is easily reached, so there is no timing issue.

The design is verified both by Spartan-3 FPGA and TSMC90 chips. There are two versions of chip implementations. In the first version, the thresholds are constant. In the second version, the thresholds are calculated on the fly. The February tape-out, which realized the first version of the subsystem, was obtained in June. The June tape-out realized the second version. In Chapter 5, the chip measurement setup and results are given. The design is functional as designed. The power consumption meets our design target and follows the power equation in digital circuit. The minimal active power achieved is  $2.1\text{ }\mu\text{w}$  at the data rate 200 kbps. To our knowledge, it is the first work on the digital implementation and chip measurement of the wakeup radio.

## 6.2 Future work

There are some recommendations for the future work:

Based on the measurement results, the error rate of the subsystem can be calculated in Matlab. To get the statistic results, Matlab has to be able to send a large amount of random packets (1,000 or even more) automatically via the FPGA and receive the corresponding wakeup signals through RS232 connection. Based on the comparison of the address and wakeup signal pattern, the error data rate is given by Matlab finally.

From the methodology point of view, the voltage scaling or multiple voltage domains can lead to lower power. From the process technology point of view, the power can be further reduced by a customized standard cells library.

In order to improve the error rate, the Manchester decoder can be optimized. Some error detection and correction techniques can be taken into consideration, such as parity bit, CRC.

# Bibliography

---

- [1] Wireless Sensors for Intelligent Transportation,  
“[http://www.cvel.clemson.edu/Projects/cvel\\_proj\\_wsn.html](http://www.cvel.clemson.edu/Projects/cvel_proj_wsn.html),” Oct. 2009.
- [2] L. Zhang, and Z. Wang, “Integration of RFID into Wireless Sensor Networks: Architectures, Opportunities and Challenging Problems,” in *Fifth International Conference on GCCW '06 on*, Oct. 2006, pp. 463-469.
- [3] F. Hu, S. Kumar, and Y. Xiao, “Towards a secure, RFID/sensor based tele-cardiology system,” in *Proceedings of the 4th Annual IEEE Consumer Communications and Networking Conference on*, Jan. 2007, pp. 732-736.
- [4] G. Coyle, L. Boydell, and L. Brown, “Home telecare for the elderly,” *Journal of Telemedicine and Telecare on*, vol. 1, pp. 183-184, 1995.
- [5] M. Ogawa, T. Tamura, and T. Togawa, “Fully automated biosignal acquisition in daily routine through 1 month,” in *International Conference on IEEE-EMBS*, Hong Kong, 1998, pp. 1947-1950.
- [6] I.A. Essa, “Ubiquitous sensing for smart and aware environments,” *IEEE Personal Communications on*, vol. 7, pp. 47-49, Oct. 2000.
- [7] C. Herring, and S. Kaplan, “Component-based software systems for smart environments,” *IEEE Personal Communications on*, vol. 7, pp. 60-61, Oct. 2000.
- [8] J.M. Rabaey, M.J. Ammer, J.L. da Silva Jr., D. Patel, and S. Roundy, “PicoRadio supports ad hoc ultra-low power wireless networking,” *IEEE Computer Magazine on*, vol. 33, pp. 42-48, 2000.
- [9] E. Shih, S. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, and A. Chandrakasan, “Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks,” in *Proceedings of ACM MobiCom '01*, Rome, Italy, July 2001, pp. 272-286.
- [10] P. Kinney, ZigBee Technology: Wireless Control that Simply Works,  
“[http://www.zigbee.org/resources/documents/ZigBee\\_Technology\\_Sept2003.doc](http://www.zigbee.org/resources/documents/ZigBee_Technology_Sept2003.doc),” Aug. 2009.
- [11] K.Y. Lin, T.K.K. Tsang, M. Sawan, and M.N. El-Gamal, “Radio-triggered solar and RF power scavenging and management for ultra low power wireless medical applications,” in *Proceedings of International Symposium on Circuits and Systems on*, May 2006, pp. 5728-5731.
- [12] V. Raghunathan, C. Schurgers, S. Park, and M.B. Srivastava, “Energy-Aware Wireless Sensor Networks,” *IEEE Signal Processing on*, vol. 19, no. 2, pp. 40-50, Mar. 2002.
- [13] C. Ho, M. Mark, M. Koplow, L. Miller, A. Chen, E. Reilly, J. Rabaey, J. Evans, P. Wright, “Technologies for an autonomous wireless home healthcare system,” in *Proceedings of the Sixth International Workshop on Wearable and Implantable Body Sensor Networks on*, June 2009, pp. 29-34.

- [14] Y.H. Chee, A.M. Niknejad, J.M. Rabaey, "An ultra-low-power injection locked transmitter for wireless sensor networks," *IEEE J. Solid-State Circuits on*, vol. 41, pp 1740-1748, Aug. 2006.
- [15] G. Walter, "Communications system for integrating a paging system with cellular radio telephones," U.S. Patent 5 541 976, June 7, 1995.
- [16] S. Misra, I. Woungang and S.C. Misra, *Guide to Wireless Sensor Networks*. London: Springer, 2009.
- [17] S. von der Mark, R. Kamp, M. Huber, and G. Boeck, "Three stage wakeup scheme for sensor networks," in *Proceedings of IEEE/SBMO International Conference on Microwave and Optoelectronics*, Brazil, July 2005, pp. 205-208.
- [18] S. von der Mark, and G. Boeck, "Ultra low power wakeup detector for sensor networks," in *Proceedings of IEEE/SBMO International Conference on Microwave and Optoelectronics*, Brazil, Oct. 2007, pp. 865.
- [19] P. Bradly, "An ultra low power, high performance medical implant communication system (MICS) transceiver for implantable devices," in *IEEE Biomedical Circuits and Systems Conference on*, Nov. 2008, pp. 32-33.
- [20] B. Van der Doorn, W. Kavelaars, and K. Langendoen, "A prototype low-cost wakeup radio for the 868 MHz band," *International Journal of Sensor Networks on*, vol. 5, no. 1, pp. 22-32, 2009.
- [21] N.M. Pletcher, "Ultra-low power wake-up receivers for wireless sensor networks," Ph.D. dissertation, Dept. EECS, University of California, Berkeley, May 2008.
- [22] Sigma-Delta ADCs and DACs, "[http://www.analog.com/static/imported-files/application\\_notes/292524291525717245054923680458171AN283.pdf](http://www.analog.com/static/imported-files/application_notes/292524291525717245054923680458171AN283.pdf)," Oct. 2009.
- [23] K. Gentile, Digital Pulse-Shaping Filter Basics, "[http://www.analog.com/static/imported-files/application\\_notes/5575241024543774944932672346AN\\_922.pdf](http://www.analog.com/static/imported-files/application_notes/5575241024543774944932672346AN_922.pdf)," June 2009.
- [24] Raised Cosine Filter, "[http://en.wikipedia.org/wiki/Raised-cosine\\_filter](http://en.wikipedia.org/wiki/Raised-cosine_filter)," June 2009.
- [25] S.W. Smith, *the Scientist and Engineer's Guide to Digital Signal Processing*. California: California Technical Pub, 1997.
- [26] A.S. Tanenbaum, *Computer Networks (4th Edition)*. New Jersey: Prentice-Hall, 2002.
- [27] G.C. Clark and J.B. Cain, *Error Correction Coding For Digital Communications*. New York: Plenum, 1981.
- [28] J. van der Tang, "Design of a generic IC test platform," Holst Centre, Eindhoven, the Netherlands, Tech. Note TN-07-WATS-TP2-013, 2007.
- [29] X. Wang and P. Harpe, "Design of PAD frame, SPI and TOP," Holst Centre, Eindhoven, the Netherlands, Tech. Note TN-09-WATS-TP2-035, 2009.
- [30] C.K. Koc and S. Johnson, "Multiplication of signed-digit numbers," *IEE Electronics Letters on*, vol. 30, no. 11, pp. 840-841, May 1994.

- [31] R.M. Hewlitt and E.S. Swartzlantler Jr. "Canonical signed digit representation for FIR digital filters," in *Proceedings of IEEE Workshop Signal Processing System on*, 2000, pp. 416.
- [32] IEEE 802.15.4a WPAN Task Group,  
["http://www.ieee802.org/15/pub/TG4a.html,"](http://www.ieee802.org/15/pub/TG4a.html) Aug. 2009.
- [33] IMEC Narrowband PHY Proposal for IEEE 802.15.6,  
["https://mentor.ieee.org/802.15/dcn/09/15-09-0340-01-0006-imec-narrowband-phy-proposal-documentation.doc,"](https://mentor.ieee.org/802.15/dcn/09/15-09-0340-01-0006-imec-narrowband-phy-proposal-documentation.doc) Oct. 2009.
- [34] F. Emmett and M. Biegel, "Power reduction through RTL clock gating," *SUNG San Jose*, 2000.
- [35] M. Dale, "Utilizing clock-gating efficiency to reduce power," *EE Times-India on*, Jan 2008.
- [36] L. Wei, Z. Chen, K. Roy, Y. Ye and V. De, "Mixed-Vth (MVT) CMOS circuit design methodology for low power applications," in *36th Annual Conference on Design Automation (DAC'99) on*, June 1999, pp.430-435.
- [37] M. de Nil, "RTL to GDS2 flow," Holst Centre, Eindhoven, the Netherlands, Tech. Note, 2008.
- [38] Spartan-3 Generation FPGA User Guide,  
["http://www.xilinx.com/support/documentation/user\\_guides/ug331.pdf,"](http://www.xilinx.com/support/documentation/user_guides/ug331.pdf) July 2009.
- [39] X. Wang, and P. Harpe, "Design of PAD frame, SPI and TOP," Holst Centre, Eindhoven, the Netherlands, Tech. Note TN-09-WATS-TP2-035, 2009.
- [40] G. Dolmans, "Description radio blocks BAN transceiver – June09 tapeout," Holst Centre, Eindhoven, the Netherlands, Tech. Note, 2009.
- [41] J.M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective (2nd Edition)*. New Jersey: Prentice-Hall, 2002.