A 0.4-V UWB Baseband Processor

Vivienne Sze
sze@mtl.mit.edu

Anantha P. Chandrakasan
anantha@mtl.mit.edu

Massachusetts Institute of Technology
50 Vassar St., Room 38-107
Cambridge, MA 02139, USA

ABSTRACT

A 0.4-V UWB digital baseband processor has been fabricated in a standard-V_T 90-nm CMOS technology. The baseband processor operates at an ultra-low supply voltage to reduce energy consumption and utilizes a highly parallelized architecture to meet throughput constraints. While ultra-low voltage operation is usually limited to low energy, low performance applications, this work examines how it can be applied to low energy, high performance applications. Measured results for a 20-pJ/bit 100-Mbps UWB baseband processor are presented. Architectural techniques and design methodologies for reducing additional complexity due to parallelism are discussed.

Categories and Subject Descriptors

B.7.1. [**Hardware**]: Types and Design Styles - *Algorithms implemented in hardware, VLSI*

General Terms

Performance, Design

Keywords

ultra-wideband, ultra-low voltage, baseband processor, parallelism

1. INTRODUCTION

Traditionally, ultra-low voltage operation has been limited to applications such as distributed sensor networks where low energy is a primary concern instead of performance. However, there are numerous applications where both maintaining high performance and lowering energy are crucial. By carefully introducing an optimum degree of parallelism to the design, we will show how aggressive voltage scaling can be applied to deliver significant energy savings without sacrificing performance and throughput.

The consumer electronics industry is exploring the use of Ultra-wideband (UWB) communications, a short-range
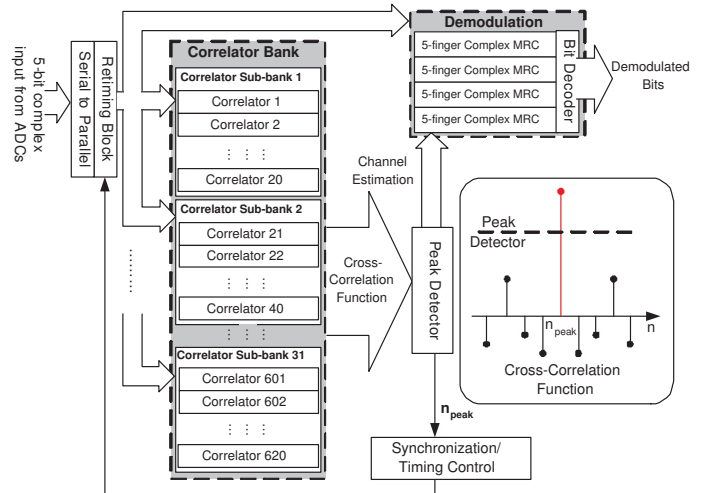
Figure 1: Architecture of highly parallelized energy efficient UWB baseband processor

high-data-rate radio technology, to complement longer range radio technologies such as Wi-Fi, WiMAX, and cellular wide area communications. UWB communications can be used to send data from a host device to other devices within the immediate area, eliminating the need for wires and increasing mobility [1]. The use of UWB as a medium for high-data-rate last-meter wireless links requires that UWB radios be integrated onto battery-operated devices such as mobile phones and handheld devices. Consequently, there is a need for an energy-efficient UWB transceiver. This work demonstrates how extreme parallelism in the digital baseband processor, shown in Figure 1, minimizes the energy required to receive UWB packets by enabling ultra-low voltage operation at 0.4 V.

This paper begins with a description of the UWB specifications and complete receiver architecture. Next, the main functions of the baseband are discussed. This is followed by a description of how parallelism can be used to achieve an energy-efficient baseband processor, how the degree of parallelism is selected, and an explanation of the design methodology used to implement it. Finally, the measured results are presented.

2. UWB SPECIFICATIONS & RECEIVER

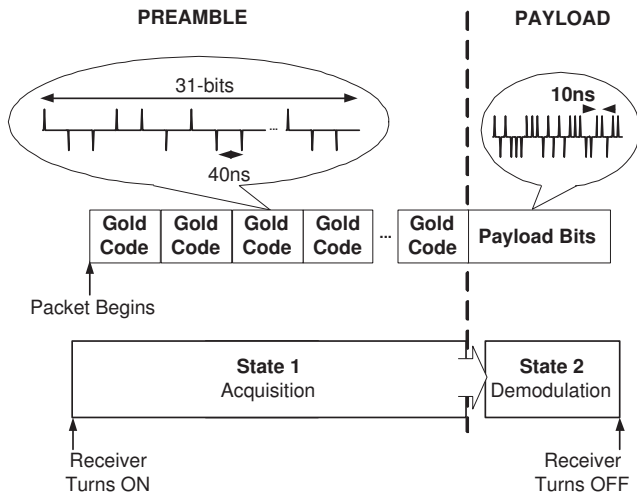The FCC has authorized UWB wireless communications

Figure 2: UWB physical-layer packet format

Figure 3: Correlator Architecture

Figure 4: Maximum Ratio Combiner Architecture. The coefficients (tap gains) are determined from the channel estimation generated by the correlator bank.

in the 3.1-GHz to 10.6-GHz band with a minimum bandwidth of 500 MHz and a maximum equivalent isotropic radiated power spectral density of -41.3 dBm/MHz [2]. There are two primary technological approaches for UWB communications: OFDM [3] and pulse-based [4]. This work focuses on the latter using 2-ns binary phase-shift keying (BPSK) pulses.

The digital baseband processor was implemented for the custom UWB receiver presented in [5]. It uses a direct-conversion architecture in the front-end and the in-phase and quadrature components of the 250-MHz down-converted pulse are sampled at a Nyquist rate of 500 MSPS by two 5-bit ADCs [6]. Consequently, for real-time demodulation of the UWB packet, the baseband processor must perform the signal processing with a throughput of 500 MSPS. Acquisition for synchronization, channel estimation and demodulation are done entirely in the digital domain and only the automatic gain control (AGC) is fed back to the analog domain. A mostly digital architecture was chosen, rather than partial analog approach [7] since it allows for deep voltage scaling and greater flexibility. The analog-to-digital conversion can be done at a relatively low power of 7.8 mW [6]. In addition, digital implementations for synchronization require shorter headers (preamble), which reduces system energy consumption [7, 8]. The baseband processor was implemented using a standard digital logic cell library in the 90-nm process.

The UWB packets are built from a sequence of BPSK pulses with a 500-MHz bandwidth. The transmitter generates approximate Gaussian pulses and up-converts the packet to one of 14 channels in the 3.1-GHz to 10.6-GHz band. The physical-layer of each packet, shown in Figure 2, can be divided into two sections: preamble and payload. The preamble contains repetitions of a N_c=31 bit Gold Code (Pseudo Noise sequence) sent at a pulse repetition frequency (PRF) of 25 MHz. The payload contains the actual data and is sent at a PRF of 100 MHz for a 100-Mbps data rate with no channel coding.
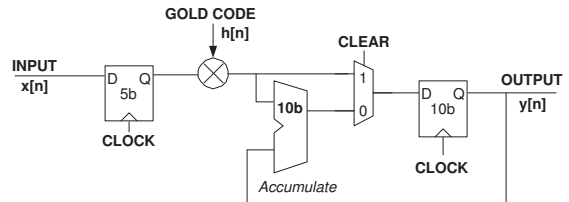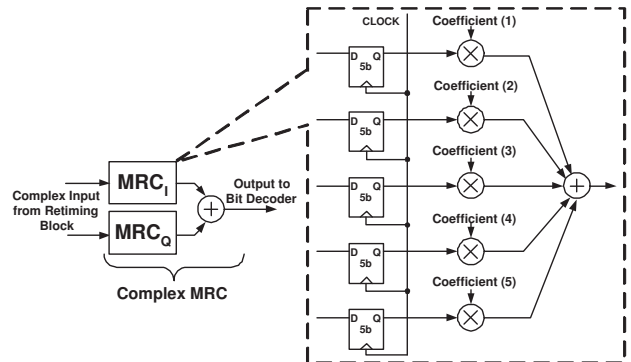
3. UWB BASEBAND PROCESSOR

3.1 Baseband Algorithm

The baseband processor implements acquisition, synchronization and demodulation by transitioning between two states of operation. The preamble is used by the receiver to achieve acquisition and synchronization. At the receiver, the baseband processor computes the cross-correlation function ($y[n]$) between the incoming noisy preamble ($x[n]$) and a clean template of the 31-bit Gold Code sequence ($h[n]$).

$$y[n] \quad = \quad \sum_{k=0}^{N_c-1} x[k] \times h[k-n]$$

The computation shown above is performed with the use of a correlator (Figure 3). It is important to note that each point of $y[n]$ can be computed independently, which makes the cross-correlation amenable to parallel processing.

Peak detection is performed on the cross-correlation to achieve signal acquisition as well as synchronization. The cross-correlation also provides the channel estimation [9]. Following synchronization, the baseband performs demodulation of the payload bits. Demodulation involves the use of a 5-fingered RAKE receiver to collect and optimally combine the signal energy received on the multiple echo paths using the tap gains determined by the channel estimation. A hard decision is made at the output of the maximum ratio combiner (MRC) to resolve a bit (Figure 4).

The total energy spent on receiving the UWB signal can be divided into two components: acquisition (preamble) energy and demodulation (payload) energy. One of the goals of this work is to reduce the energy spent by the receiver on acquisition. Since this energy does not go directly to-
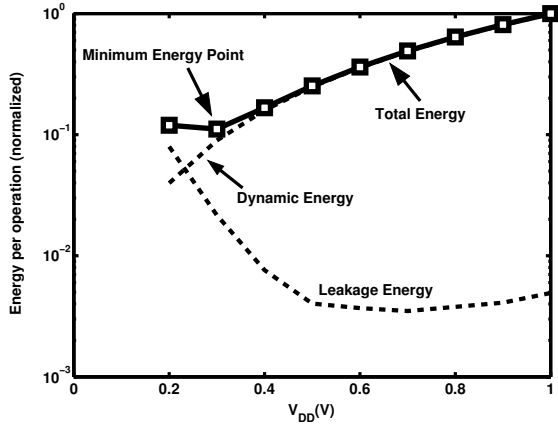
Figure 5: Simulated energy plot for the correlator.

ward the demodulation of the data, it is seen as overhead energy. During short bursty traffic, where the payload is small, this overhead energy accounts for a significant portion of the total packet energy. Therefore, it is desirable to minimize the amount of overhead energy per packet. The majority of this overhead energy goes into the computation of the cross-correlation function. This overhead energy can be significantly reduced by exploiting the use of parallelism.

3.2 Ultra-Low Voltage Operation

There are a fixed number of operations required by the baseband in order to compute the cross-correlation function, and therefore in order to reduce the energy of the baseband, we need to reduce its energy per operation. This is achieved by scaling down the supply voltage (V_{DD}) such that the correlator, which computes the cross-correlation, operates at its minimum energy point [10]. The minimum energy point occurs since the total energy per operation is composed of dynamic energy and leakage energy.

$$
\begin{aligned}
E_{total} &= E_{dynamic} + E_{leakage} \\
&= C_{eff}V_{DD}^2 + I_{leak}V_{DD}T_{delay}
\end{aligned}
$$

From the total energy equation we see that lowering V_{DD} decreases the dynamic energy. While reducing V_{DD} reduces the leakage power, it also increases the delay (T_{delay}) of the gates. When the V_{DD} is above the threshold voltage of the device, the delay increases linearly with V_{DD}, and there is no significant change in the leakage energy; however, when V_{DD} drops below the threshold voltage of the device, both delay and leakage energy increase exponentially. Since the dynamic energy and leakage energy scale in opposite directions as V_{DD} decreases, a minimum energy point occurs near the sub-threshold region. Spectre simulations performed on the correlator indicate that the minimum energy point occurs at 0.3 V, which gives a 9× energy reduction as compared to the full-scale 1-V operation (Figure 5). Ideally, it would be desirable to scale V_{DD} such that the baseband operates at this minimum energy point.

However, as previously mentioned, the baseband processing must sustain a throughput of 500 MSPS in order to achieve real-time demodulation. This can be achieved by a single correlator operating at a frequency of 500 MHz with a much higher voltage than 0.3 V, but we have shown that

this is not energy efficient. Instead, it is better to operate at an ultra-low voltage at a reduced frequency, and utilize parallelism in the baseband to meet the throughput constraint.

In order to refrain from introducing additional complexity due to parallelism, it is preferable that the operating frequency be a factor of the preamble PRF (25 MHz). This allows for an integer number of pulses to be processed every cycle. The operating frequency is equal to 25 MHz if the supply voltage is raised slightly to 0.4 V. Since the minimum energy point is shallow, this slight change in V_{DD} does not cause a significant energy penalty. By operating at 0.4 V rather than 1 V, the energy per operation is reduced by almost 6×. At 25 MHz, the correlator needs to be parallelized by a factor of 20 in order to maintain the 500-MSPS throughput.

In summary, the method of selecting the optimal degree of parallelism to minimize energy consumption, while meeting performance constraints, involves the following three steps:

1. Determine the V_{DD} at which the block (e.g. correlator circuit) operates at its minimum energy point

2. Determine the delay and throughput of the block at this V_{DD}

3. Divide the required throughput by the throughput of the block operating at V_{DD} to obtain the necessary degree of parallelism

This form of parallelism can also be used to reduce the energy spent on the demodulation of the payload bits. The MRC of the RAKE receiver is parallelized by a factor of 4 such that it can operate off the same supply voltage and operating frequency as the rest of the baseband. Combining parallelism with ultra-low voltage operation delivers energy savings for receiving the entire UWB packet.

In addition to lowering the energy of the baseband processor, the overhead energy spent by the other blocks in the receiver (RF front-end, ADCs and baseband amplifiers) should be reduced. This is achieved by reducing the acquisition time, such that the overall on-time of the entire receiver is shorten, and applying duty-cycling.

Reduced acquisition time can be achieved by further parallelizing the correlator to compute multiple points in the cross-correlation function at the same time. Assuming that the communication protocol is flexible, this results in fewer Gold Code repetitions and thus shorter packets, which translates to shorter receiver on-time. When the baseband processor is parallelized by N_C=31, all points of the cross-correlation function can be computed simultaneously. The overhead energy of the entire receiver is lowered by 14.7×, based on the measured results of the other blocks in the receiver [6, 11] and assuming $P_d = 0.9$ and $P_{fa} = 10^{-5}$. Further analysis is presented in [12].

3.3 Baseband Architecture

The combination of these two approaches results in a highly parallelized implementation with a total of 620 correlators and 4 RAKE MRCs. The parallelized architecture is shown in Figure 1. There are 20 correlators in each sub-bank to maintain the 500-MSPS throughput, and 31 sub-banks so that all points of the cross-correlation can be computed simultaneously. The first form of parallelism, which reduces the energy of the baseband processor, is determined by the frequency of the correlator near its minimum energy point,
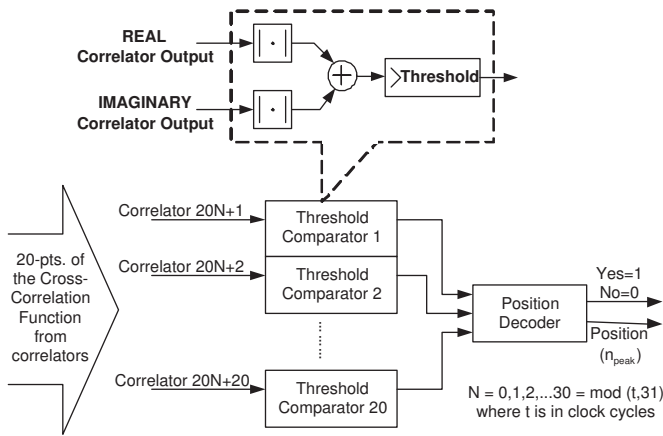
Figure 6: Peak detector reuses 20 threshold comparators for 620 correlators

Figure 7: Energy-Area trade-off for baseband

while the second form, which reduces the energy of the other blocks in the receiver, is dictated by the length of the Gold Code sequence (N_c).

In highly parallelized designs, blocks should not be blindly replicated as this could result in large unnecessary area penalties. Parallelism must be applied in an efficient manner to minimize the associated hardware costs. Thus, we want to exploit the maximum amount of block reuse. This methodology is demonstrated by the co-design of the correlator bank and the peak detector. The peak detection is used to search for the maximum of the cross-correlation which indicates when the input and template are aligned. The cross-correlation between the incoming signal and the Gold Code can be computed by the parallelized correlator bank in two ways: either the same input goes to all correlator sub-banks and each sub-bank contains a template of the Gold Code with a different delay, or all sub-banks contain the same Gold Code template and each sub-bank receives the incoming signal with a different delay. The latter implementation is selected since each sub-bank produces outputs at staggered clock cycles. This allows for the threshold comparators in the peak detector, shown in Figure 6, to be time-shared by all 620 correlators. Only 20 threshold comparators are required to service the output of the 20 correlators per sub-bank each cycle, as compared to the case where 620 threshold comparators would be required if all the correlators produced outputs in the same clock cycle. This provides 31× savings in terms of both area and leakage energy.

Note that time-sharing is made possible by the fact that the peak detection involves comparing the output of the correlator to a threshold value rather than doing a search for the true maximum. The threshold comparisons are done in chronological order (i.e. threshold detection will detect lock in earlier positions than later ones). Although, there is the possibility that the position at which the correlator output exceeds the threshold is not at the maximum of the cross-correlation function, the threshold can be selected such that there is a high probability that only the maximum is detected. The threshold is made programmable so that it can be adjusted during testing. This modification to the algorithm was verified via MATLAB simulation as well as a system prototyping platform [13].
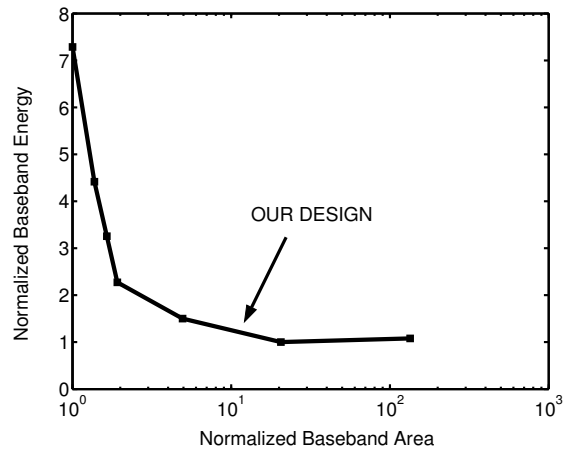
4. ENERGY-AREA TRADE-OFF

The previous sections describe how parallelism can be used to reduced energy at the cost of increased area. In this section, we will quantify these costs. In addition to area increase from replication of the correlators and the threshold comparators, there is overhead area due to the need for additional multiplexers, specifically the muxes located at the output of the correlators, and two serial-to-parallel blocks at the I/Q inputs for parallel operation. By processing an integer number if pulses per cycle, this overhead is kept at a minimum.

Understanding and quantifying this explicit trade-off between energy and area allows one to optimize the design for both energy and area. Figure 7 shows the trade-off curve between the baseband processor energy and the area of the baseband processor. In our design, the baseband energy is reduced by a factor of 6× at the cost of a 9.6× increase in area.

5. METHODOLOGY & IMPLEMENTATION

The parallelized baseband algorithm was first verified using MATLAB to ensure correct functionality. This setup was also useful in generating test vectors. Initially, only the correlator was synthesized by Design Compiler using the 90-nm standard cell library. Spectre was then used to simulate the correlator to determine its minimum energy point. The standard cell library was re-characterized with Signal-Storm for the optimum voltage point of 0.4 V. At ultra-low voltages, the delay of the gates decreases with temperature, which is contrary to the behavior in full scale operation. This is because I_{off} increases with temperature, while I_{on} decreases with temperature. The corner library characterizations take this into account (i.e. the fast corner used a higher temperature than the slow corner).

With the use of Perl scripting, the baseband algorithm was translated into digital circuits written in Verilog with the appropriate degree of parallelism. The entire baseband processor was then synthesized with the 0.4-V library, and Astro was used for place-and-route. Distributed clock gating was incorporated for further power savings on the clock network. For instance, this ensures that the large correlator bank is not clocked during demodulation.

CHIP FEATURES	
Technology	90-nm CMOS
Core area	2.9 mm × 2.9 mm
Frequency	25 MHz
Cell count	281,260
IO / core V_{DD}	1 V / 0.4 V
Power (acq/dem)	7 mW / 1.7 mW

Figure 8: Die photo of baseband processor and summary of chip features

Also, due to the high degree of parallelism, a hierarchal approach was used to minimize the turn-around time of the EDA tools. Synthesis was performed first on a single correlator, followed by the correlator sub-bank, the entire correlator bank and finally the top-level baseband processor.

Timing verification that incorporated global variations was performed using PrimeTime. At ultra-low voltages, the impact of local "transistor-to-transistor" variations is quite severe. Consequently, Monte Carlo simulations were performed for additional variation analysis using Spectre to verify timing on critical paths. The circuit was also simulated in Nanosim with extracted RC parasitics.

A serial-to-parallel converter was required at the input of the baseband processor to reduce the number of I/O pads resulting in a second clock domain of 100 MHz. The timing constraints for signals crossing the clock domains had to be carefully set and verified. Note that if the receiver was implemented as a system on a chip, the serial-to-parallel block would not be required if the ADC was also parallelized.

6. EXPERIMENTAL RESULTS

6.1 Measured Results

The baseband processor implemented in a standard-V_T 90-nm CMOS process, shown in Figure 8, demonstrates 100-Mbps operation at 0.4 V with an operating frequency of 25 MHz. A summary of the performance metrics is also shown in Figure 8. A scope plot of the baseband operating at 0.4 V is shown in Figure 9. Only 23% of the die area is active; the rest is filled with decoupling MOS capacitors. The active area of the baseband is comparable to the total active area of the RF front-end and ADC [6, 11].

The breakdown of the energy per bit consumed by the baseband is shown in Figure 10. The average overhead energy consumed during acquisition is fixed for a packet. Thus, the shorter the payload, the greater the overhead energy per bit as the overhead energy is amortize over few bits. For a 4-kbit packet, which is within the limits of the allowed 802.11 PSDU length (i.e. payload) [14, 15], the average energy per bit consumed by the baseband processor is 20 pJ with 3 pJ going toward acquisition and 17 pJ going to demodulation.

A direct comparison of energy number with similar work [16, 17, 18] is difficult due to different system design parameters such as data rate, resolution of input bits, spread sequence length, equalization, signal bandwidth, etc. [16] is designed for a 4-bit UWB baseband signal in the 0 - 300 MHz band. [17] operate in the 0 - 960 MHz band and takes in 1-bit inputs. [18] is the most similar to our design, operating on the same 500 MHz bandwidth signal with direct conversion (I/Q) from the 3.1 - 10.6 GHz band and the same
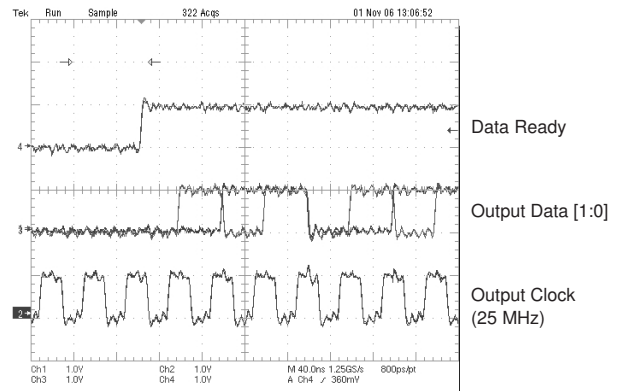

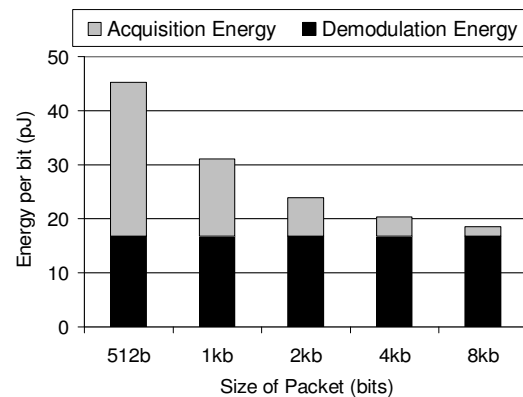**Figure 9: Scope plot showing correct functionality at 400 mV. Note: I/O has a 1 V power supply**

Figure 10: Breakdown of energy per bit consumed by baseband

5-bit input resolution. It also contains equalization, however the energy consumed by the MLSE equalizer is not included in the energy comparison. For similar packet length conditions, the pulse-based UWB digital baseband processors consume 12.5 nJ, 107 pJ and 847 pJ respectively, while our implementation consumes 20 pJ. By operating at an ultra-low voltage, the baseband processor achieves energy savings as compared with current state-of-the-art UWB baseband transceivers.

6.2 Power Gating

Both forms of parallelism assume that the receiver can be powered off. Off-chip power gating was used to demonstrate this with the baseband processor (Figure 11). Power gating involves gating the leakage current when the system is idle. A Fairchild NFET was used as the gating transistor.

Realistically, power gating itself costs energy; specifically, the energy required to switch the gating transistor and the recovery energy required to bring the virtual V_{DD} back up to 0.4 V. There is a minimum amount of time that the system must be powered off in order for power gating to be advantageous. This time, known as the break-even time, occurs when the savings in leakage energy is greater than the cost of power gating. The break-even time for the baseband proces-
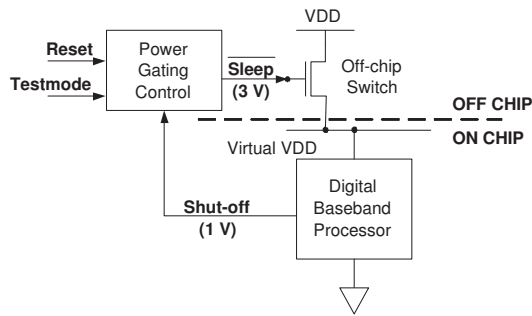
Figure 11: Off-chip Power Gating

sor was determined to be 137 μs. A shut-off signal for power gating is automatically generated by the baseband when the packet is completely demodulated. The turn-on signal could be generated at a higher level (e.g. MAC layer).

Off-chip digital gates were used to implement the power gating control logic. The off-chip gating transistor has a 3-V switching voltage, which required that the control logic operate at 3 V, and a level converter be used to interface the control logic with the 1-V shut-off signal from the baseband processor I/O. A separate 3-V supply voltage was used to power this off-chip control logic.

7. CONCLUSIONS

This paper examined how extreme parallelism can be exploited to enable the use of ultra-low voltage operation as a viable energy savings technique for high performance applications. Furthermore, for highly parallelized implementations, maximizing block reuse and minimizing the overhead and complexity due to parallelism are keys to obtaining an energy-efficient design. Challenges pertaining to highly parallelized designs with ultra-low voltage operation such as tool turn-around time and sensitivity to variations must be addressed throughout the design process. The analysis in this paper can be mapped to other high performance communication applications using ultra-low voltage operation and parallelism.

8. ACKNOWLEDGEMENTS

9. REFERENCES

[1] "Ultra-Wideband (UWB) Technology, Technology & Research at Intel," http://www.intel.com/technology/comms/uwb/.

[2] "Ultra-wideband First Report and Order," Tech. Rep., Federal Communications Commission, February 2002.

[3] A. Batra, et al., "Multi-band OFDM physical layer proposal for IEEE 802.15 Task Group 3a," Tech. Rep., IEEE P.802.15-04/0493r0, September 2004.

[4] R. Fisher, et al., "DS-UWB physical layer submission to 802.15 Task Group 3a," Tech. Rep., IEEE 802.15-04/0137r3, July 2004.

[5] F. S. Lee, R. Blazquez, B.P. Ginsburg, J.D. Powell, M. Scharfstein, D.D. Wentzloff, and A.P. Chandrakasan, "A 3.1 to 10.6 GHz 100 Mb/s Pulse-Based Ultra-Wideband Radio Receiver Chipset," in *IEEE International Conference on Ultra-Wideband*, September 2006.

[6] B. P. Ginsburg and A.P. Chandrakasan, "Dual Scalable 500MS/s, 5b Time-Interleaved SAR ADCs for UWB Applications," in *IEEE Custom Integrated Circuits Conference*, September 2005.

[7] M. Verhelst and W. Dehaene, "System Design of an Ultra-low Power, Low Data Rate, Pulse UWB Receiver in the 0-960MHz Band," in *IEEE International Conference on Communications*, May 2005.

[8] J. Ammer and J. Rabaey, "Low Power Synchronization for Wireless Sensor Network Modem," in *IEEE Wireless Communications and Networking Conference*, March 2005.

[9] R. Blazquez and A. P. Chandrakasan, "Architectures for Energy-Aware Impulse UWB Communications," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2005.

[10] B. H. Calhoun and A. P. Chandrakasan, "Characterizing and Modeling Minimum Energy Operation for Subthreshold Circuits," in *International Symposium on Low Power Electronics and Design*, August 2004.

[11] F. S. Lee and A.P. Chandrakasan, "A BiCMOS Ultra-Wideband 3.1-10.6GHz Front-End," in *IEEE Custom Integrated Circuits Conference*, September 2005.

[12] V. Sze, R. Blazquez, M. Bhardwaj, and A.P. Chandrakasan, "An Energy Efficient Sub-Threshold Baseband Processor Architecture For Pulsed Ultra-Wideband Communications," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2006.

[13] D.D. Wentzloff, R. Blazquez, F.S. Lee, B. P. Ginsburg, J. Powell, and A.P. Chandrakasan, "System Design Considerations for Ultra-Wideband Communication," *IEEE Communications Magazine*, vol. 43, no. 8, pp. 114–121, August 2005.

[14] J. Yin, X. Wang, and D.P. Agrawal, "Optimal Packet Size in Error-prone Channel for IEEE 802.11 Distrbuted Coordination Function," in *IEEE Wireless Communications & Networking Conference*, 2004.

[15] "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer specifications (PHY), High-speed Physical Layer in the 5-GHz Band," Tech. Rep., IEEE Std 802.11a-1999(R2003).

[16] R. Blazquez, P.P. Newaskar, F.S. Lee, and A.P. Chandrakasan, "A Baseband Processor for Impulse Ultra-Wideband Communications," *IEEE Journal Of Solid-State Circuits*, vol. 40, no. 9, pp. 1821–1828, September 2005.

[17] C.-H. Yang, K.-H. Chen, and T.-D. Chiueh, "A 1.2V 6.7mW Impulse-Radio UWB Baseband Transceiver," in *IEEE International Solid-State Circuits Conference*, February 2005.

[18] Raul Blazquez, *Ultra-wideband Digital Baseband*, Ph.D. thesis, Massachusetts Institute of Technology, May 2006.