Design of a Low-Power Wireless Camera

Anantha Chandrakasan, James Goodman, James Kao, Wendi Rabiner, Thomas Simon
Department of EECS,
Massachusetts Institute of Technology, Cambridge

ABSTRACT

This paper describes the system design of a low-power wireless camera. A system level approach is used to reduce energy dissipation and maximize battery lifetime. System properties such as the network configuration and data statistics are exploited to minimize computational switching. Embedded power supplies systems are also used to minimize energy dissipation under varying temperature, process parameters and computational workload. Since sensor systems often operate in burst mode with long idle periods, emphasis must be placed on reducing system leakage power through the use of emerging technologies and circuit techniques.

1. Introduction

This paper describes the design considerations for an ultra low-power wireless camera (Figure 1). The camera transmits compressed video data over a wireless link (with a variable bandwidth up to 1Mbps) to a fixed base station. Many of the design issues faced in the context of our wireless camera are common to those in other wireless applications. Total system energy (computation and communication) averaged over the normal operating conditions of the device should be minimized to maximize battery lifetime. The system should also be designed to service time varying data rates and quality of service requirements; embedded power supplies, which adapt supply voltages on demand, can save significant power in such systems.

The main application specific issues of our camera have to do with the asymmetry between a camera and the receiving base-station, which may be communicating with several cameras. The base-station is assumed to not be battery operated and this creates opportunities to design a system with skewed computational burdens, favoring the low-power cameras.

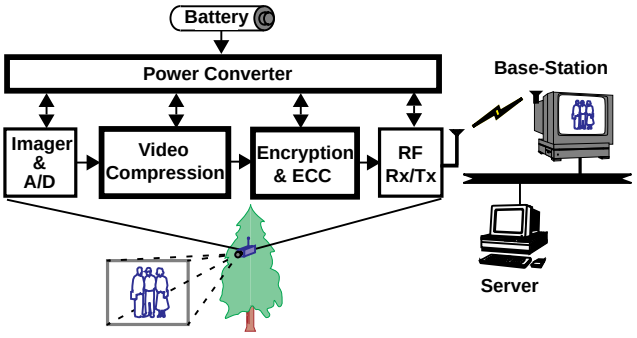

Figure 1. Portable video camera in a networked environment

2. Network-Driven Optimization

Increasingly, portable devices are being operated within a networked environment, which gives them access to high powered compute servers through a combination of wired and wireless channels. This environment with distributed resources can be exploited to reduce the power dissipation in the portable by transferring large amounts of computation typically performed at the portable to the high powered servers on the network.

The camera described in the previous section must encode the video stream before it can be transmitted across the bandwidth-constrained wireless channel. Most video compression algorithms use some form of block-based scene motion estimation/compensation to remove the temporal correlation inherent in natural video sequences. The algorithms vary from low-complexity approaches such as conditional replenishment (i.e., simple frame differencing) to full search approaches. There is an inherent trade-off between the amount of computation and the rate of compression for conventional video compression algorithms-- the ones which achieve high rates of compression are computationally intense, whereas the computationally simple algorithms tend to not produce acceptable rates of compression.

An important observation, that can save significant power in the camera, is that the motion of objects is continuous from one frame to the next in natural sequences. Thus, knowing the location of an object in a few previous frames, it is possible to predict its location in the current frame (i.e., it is possible to predict the motion vectors of the current frame based on the motion vectors of the previous frames). It therefore is possible to remove the motion estimation computation from the portable encoder and perform it at the receiving base-station (or at another server on the network). Since the base-station only has access to the previous reconstructed frames, it must perform motion estimation on these previous frames and predict the motion vectors of the current frame from these motion vectors of the previous reconstructed frames. These predicted motion vectors are transmitted through a low-bandwidth reverse wireless link to the encoder, where

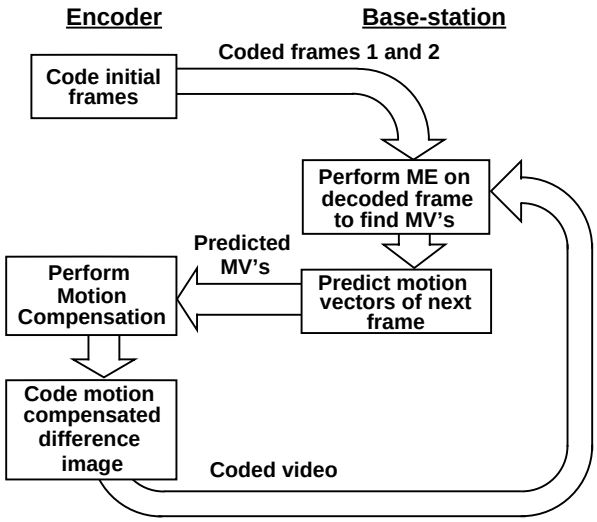

Figure 2. Network-Driven Motion Estimation

the remainder of the compression algorithm is performed. This algorithm, shown in the flow graph in Figure 2, is called network-driven motion estimation and illustrates the effective reworking of a conventional video compression algorithm so that the computationally demanding task of motion estimation can be performed at a high powered network server [1].

A video compression system which employs network-driven motion estimation performs nearly as well as one which performs the motion estimation computations at the encoder using full search computation. This is illustrated in Figure 3, which shows the number of bits required to code each frame of a video sequence using encoder-based motion estimation, conditional replenishment, and network-driven motion estimation. Network-driven motion estimation compresses the images much more than conditional replenishment, while using the same amount of encoder power. Alternatively, network-driven motion estimation achieves nearly the same compression rates as encoder-based motion estimation while

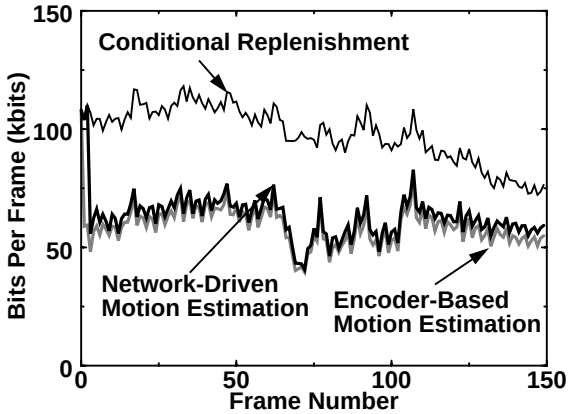

Figure 3. Bit rates for different video algorithms

using over *two orders of magnitude* less computation at the encoder.

3. Energy Scalable Computing

Energy efficient system design requires systematic optimization at all levels of design abstraction ranging from process technology and logic design to architectures and algorithms [2]. In many applications, such as the wireless camera, it is desirable to design digital processors that allow a trade-off between the quality of service (QoS) provided and the energy consumed to process a sample. This allows the user to evaluate the application's requirements and set the desired quality while minimizing the energy consumption. We have developed an energy scalable encryption processor where the level of security (i.e., quality) and energy consumed to encrypt a bit can be traded-off dynamically based on demand. Since transmitted data streams can often be partitioned into different priority levels, an energy scalable processor ensures that important information is adequately protected, while sacrificing some security for low priority data in order to reduce the total system energy.

The energy scalable encryption processor in this work is based on a variable-width quadratic residue generator (QRG). The QRG is a cryptographically-secure pseudo-random bit generator that is based upon the work in [3]. The QRG operates by performing repeated modular squarings. The modular squaring is performed using an algorithm based on Takagi's iterated radix-4 algorithm [4] which requires $(\log_2 Q)/2$ iterations to compute the result $P = X \cdot Y \bmod Q$. The least significant $\log_2 \log_2 Q$ bits of each result can be extracted and used as a strong reproducible pseudo-random source for applications such as a stream cipher or key generator.

Energy scalable computing requires dynamically reconfigurable architectures that allow the energy con-

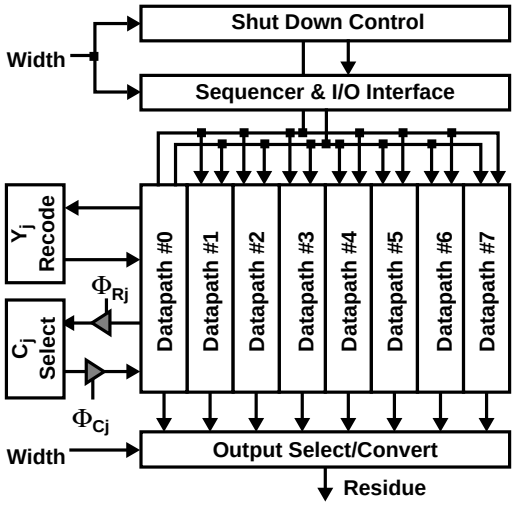

Figure 4. Energy scalable encryption processor

sumption per input sample to be varied with respect to quality. In the case of the QRG, the quality scales sub-exponentially with the modulus length, while the energy consumption scales polynomially. A fully scalable QRG architecture was developed where the width ($w = \log_2 Q$) can be reconfigured on the fly to range from 64 to 512 bits in 64 bit increments (Figure 4) [5]. The design makes extensive use of clock gating to disable unused portions of the QRG. Hence the switched capacitance of the QRG is minimized and energy scalability is achieved.

Further energy/security scalability can be achieved through the use of an adaptive supply [6]. Rather than designing a system with a static supply to meet a specific timing constraint under worst case conditions, it is more energy efficient to allow the voltage to vary such that the timing constraints are just met at any given temperature and operating conditions; this is accomplished by establishing the power supply feedback around a fixed processing rate or delay. In this example, when operating at a reduced width, the number of cycles required per multiplication is reduced and therefore the supply voltage can be reduced for a given throughput. The supply is varied using an embedded custom DC/DC converter. The use of an adaptive supply enables us to substantially reduce the energy consumption as the multiplier width is varied (Figure 5). Figure 6 shows a plot of security (measured in MIPS-years) as a function of energy per bit. This plot was obtained by varying the bitwidth and supply. Figure 7 shows a die photo of the scalable encryption processor with embedded power supply.

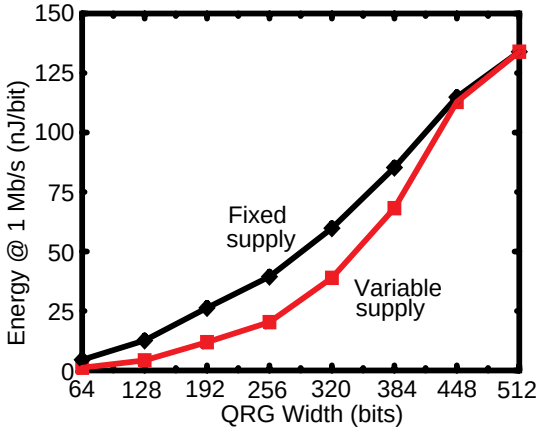

Figure 5. Power dissipation for fixed and variable supply

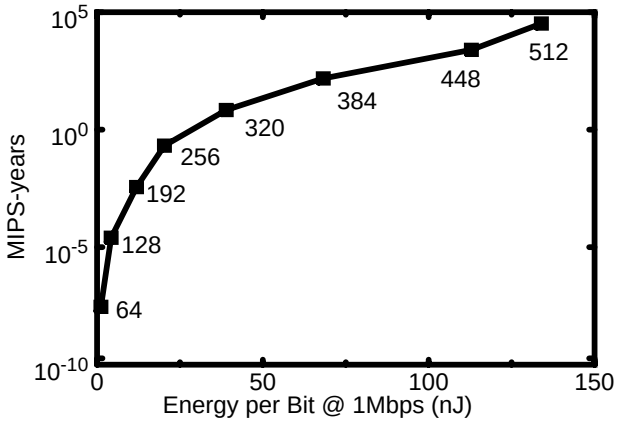

Figure 6. Security vs. Energy

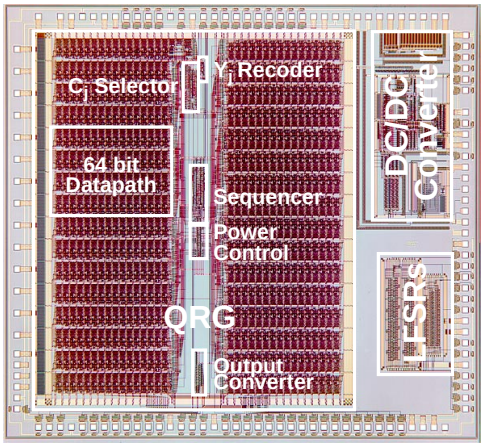

Figure 7. Encryption processor with embedded supply

The idea of varying energy and quality is a general concept that can be applied to other signal processing modules in the camera. A good example is the video compression algorithm, where the image quality (amount of compression) and energy to encode a frame can be traded-off. Using a differential wavelet compression algorithm, as shown in Figure 8, energy/quality scalability is obtained by coding each frame with a variable number of bits.

Figure 8. Image at various compression ratios

4. Burst Mode Computing

Sensor systems should also be designed to deal with low duty cycles. In the wireless camera application, there can be extended periods when there is very little image motion. During periods of low motion, the signal process-

ing and transmission circuitry can be powered off. In order to minimize energy in the sleep state, device leakage must be minimized. However, low-voltage/low-power design requires the use of low threshold devices which results in significant static power dissipation.

One emerging technology that addresses this problem is Multiple Threshold CMOS (MTCMOS), which reduces leakage current during idle mode by providing a high threshold "gating" transistor in series with the low V_t circuit transistors (Figure 9). In active mode, the high V_t transistor is turned on, while in sleep mode it is turned off, providing a small subthreshold leakage current [7]. For a purely combinational circuit, where state does not need to be preserved, only one type of high V_T device is actually required.

A key challenge in designing with MTCMOS is the transistor sizing of high V_T devices, which is very strongly dependent on data dependent current profiles. As logic gates switch, the virtual power lines fluctuate, causing logic to slow down (due to body effect and reduced current drive); this makes analyzing the critical path a challenging task. For example, consider two vector pairs (A, B) that exercise the same critical path and hence result in the same delay for an 8x8 multiplier in standard CMOS technology (Figure 10) [8]. The plot shows the delay for an equivalent MTCMOS implementation for the two different vectors pairs. The delay for a given sleep transistor size differs because the transition from (x:00,y:00) -> (x:FF,y:81) causes many more internal transitions in adjacent cells and thus is more susceptible to ground bounce than the (x:7F, y:81) -> (x:FF, y:81) transition. The second

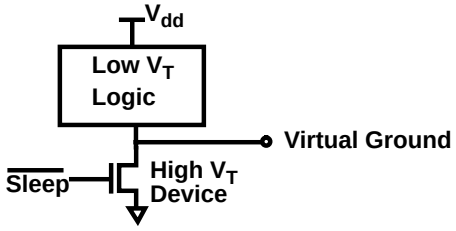

Figure 9. MTCMOS circuit structure

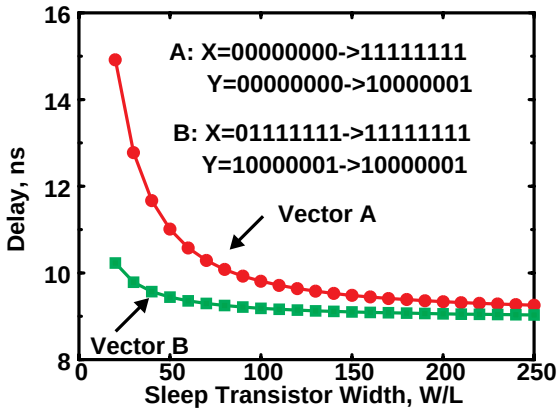

Figure 10. MTCMOS delay dependence on activity

input causes a rippling effect through the multiplier, where only a few blocks are discharging current at the same time. New design methodologies and tools will be required to analyze critical paths and optimize transistor sizes in MTCMOS technology.

5. Conclusion

Low-power sensor design requires a system level methodology that explicitly considers computation and communication costs. Networking and remote compute servers can be exploited to save power by optimum partitioning of the computation. Extremely low power operation can be achieved in digital circuits by aggressively scaling of the power supply voltage. In many cases, the supply has to be adaptive to meet time varying QoS or data rate requirements. The low duty cycle of sensors should be considered in the selection of process technology and circuity styles since idle mode leakage can dominate the overall power consumption.

Acknowledgments

This work is sponsored by DARPA, under contracts #DAAL-01-95-K3526 and contract #DABT63-95-C-0088. We thank Abram Dancy for the design of the DC-DC converter circuitry.

References

[1] W. Rabiner and A. Chandrakasan, "Network-Driven Motion Estimation for Wireless Video Terminals," IEEE Transactions on Circuits and Systems for Video Technology, pp. 644-653, Aug. 1997.

[2] A. Chandrakasan, R. Brodersen, *Low-Power CMOS Design*, IEEE reprint book, Jan. 1998.

[3] L. Blum, M. Blum, and M. Shub, "A Simple Unpredictable Pseudo-Random Number Generator," SIAM Journal on Computing, v. 15, n. 2, pp. 364-383, 1986.

[4] N. Takagi, "A Radix-4 Modular Multiplication Hardware Algorithm for Modular Exponentiation", IEEE Transactions on Computers, pp. 949 - 956, Aug. 1992.

[5] J. Goodman, A. P. Chandrakasan, "A 1Mb/s Energy/Security Scalable Encryption Processing Using Adaptive Width and Supply," IEEE International Solid-state Circuits Conference, pp. 110-111, Feb. 1998.

[6] V. Gutnik, A. Chandrakasan, "Embedded Power Supply for Low-Power DSP," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, pp. 425-435, Dec. 1997.

[7] S. Mutoh, T. Douseki, Y. Matsuya, T. Aoki, S. Shigematsu, J. Yamada, "1-V Power Supply High-Speed Digital Circuit Technology with Multithreshold-Voltage CMOS," *IEEE JSSC*, vol. 30, no. 8, pp. 847-854, August 1995.

[8] J. Kao, A. Chandrakasan, D. Antoniadis, "Transistor Sizing Issues and Tool For Multi-Threshold CMOS Technology", *IEEE/ACM DAC*, pp.409-414, June 1997.