Conclusions: The test results show that the filters in the input and output stages of the module ensure isolation such that the dual band power amplifier may operate successfully at two frequency bands. Using the new design method, the maximum total operating current was successfully reduced to 120mA which is much lower than the value (~300mA) of commercial products for CDMA application. The module size can be miniaturised to be as small as 0.96CC, due to using multilayer and small size chip components.

3 November 1997
Electronics Letters Online No: 19981448

Gi-Ho Yun (*Department of Radio Communication Engineering, Honam University, 59-1 Seobong-Dong, Kwangsan-Gu, Kwangju-City, 506-090 Korea*)

E-mail: ghyun@honam.honam.ac.kr

Hyun-Goo Yoon and Han-Kyu Park (*Department of Radio Communication Engineering, Yonsei University, 134 Shinchon-Dong, Seodaemun-Ku, Seoul, 120-749 Korea*)

References

1 NAGAOKA, M., and INOUE, T.: 'A monolithic GaAs linear power amplifier operating with a single voltage low 2.7V supply for 1.9GHz digital mobile communication applications', *IEEE Trans. Electron.*, 1995, **E78-C**, (4), pp. 424–429

2 YOKOYAMA, T., KUNISHISA, T., and FUJIMOTO, H.: 'High efficiency low adjacent channel leakage GaAs power MMIC for 1.9GHz digital cordless phones', *IEEE Trans. Microw. Theory Tech.*, 1994, **MTT-42**, pp. 2623–2628

3 STAUDINGER, J.: 'Improving amplifier linearity with harmonic load & source term. for two tone sinusoidal stimuli'. 1997 IEEE MTT-S Topical Symposium on Tech. for wireless Application, Vancouver, BC, Canada

Low power parallel Huffman decoding

Chia-Hsing Lin and Chein-Wei Jen

A low power design technique for a parallel Huffman decoder is presented. According to the length of the incoming Huffman codeword, the proposed strategy makes the barrel shifter in the parallel Huffman decoder turn off the unnecessary shifting bits to reduce power dissipation. The result of a SPICE simulation indicates that up to 50 percent power reduction in the barrel shifter may be achieved with the proposed technique.

Introduction: Huffman coding is a lossless data compression technique widely adopted in image and video coding applications like JPEG and MPEG. The main idea of the Huffman code is to use variable-length codewords to compress input symbols based on the probability of the symbol's occurrence. The most frequently used symbols are represented with shorter codes, while less frequently occurring symbols have a longer representation. The codewords assigned to input symbols will follow the prefix condition property whereby no codeword is a prefix or an initial part of another codeword.

For applications with high throughput requirements, such as high definition television (HDTV), the parallel Huffman decoder architecture was proposed [1, 2] to provide the constant-output-rate Huffman decoding. The parallel Huffman decoder mainly consists of two units: a barrel shifter for the alignment of incoming Huffman codeword and a lookup table for generation of output symbols and codeword length. Both of these two units will consume significant energy during operation of a parallel Huffman decoder. In [3] Cho *et al.* proposed the approach of non-uniform table partitioning to reduce the power dissipation of the lookup table for Huffman decoding. In this Letter, we present a power minimisation strategy for the barrel shifter to further reduce the power consumption of the parallel Huffman decoder.

Parallel Huffman decoder: Fig. 1 shows the general architecture of the parallel Huffman decoder [1, 2]. The barrel shifter stores a $2M$ bit window of the input data, where the lookup table (LUT) takes an M bit slice as its input to decode the output symbol and codeword length. The length is then fed to the accumulator to control the selection of the M bit slice from the $2M$ bit window for the barrel shifter. The input bus of the barrel shifter can be connected to the output bus via the pass transistors, where the control signal S_n ($1 \le n \le M$) selects the pass transistors to be switched. The barrel shifter is not a critical unit for the delay. However, it does consume significant energy while selecting another bit slice because of the high switching capacitance associated with the control line and the input/output bus. The power consumption of the barrel shifter is in proportion to the width of the output bit slice.

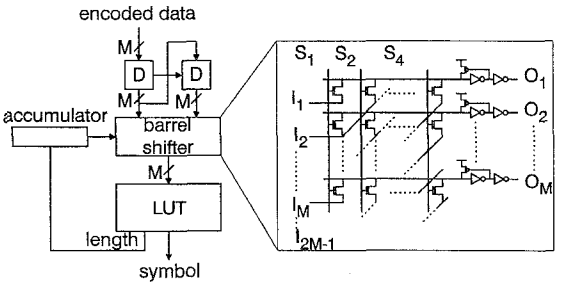

Fig. 1 *General parallel Huffman decoder*

Because the codeword length is variable, not all the bits in the M bit slice are always needed for the LUT to decode the codeword. That is, if the length of the current codeword is N, the other (M–N) bits in the bit slice do not in fact need to be shifted out by the barrel shifter. The LUT will be unconcerned with the last (M–N) bits on the output bus because there is no Huffman codeword that is a prefix of another codeword. Furthermore, shorter codewords usually occur more frequently than the longer ones in Huffman coding in order to achieve the data compression. Therefore, if we can find a way to shift out only the necessary bits to the LUT while preserving the states of the unused bits, it is possible to reduce the power dissipation of the barrel shifter.

Low-power parallel Huffman decoder: Fig. 2 shows the proposed barrel shifter architecture with a power reduction technique for parallel Huffman decoding. The original control line S_n ($1 \le n \le M$) in the barrel shifter is now split to K-way line segments S_n^1, S_n^2, ...S_n^K and each signal S_n^l controls M^l bit shifting of the barrel

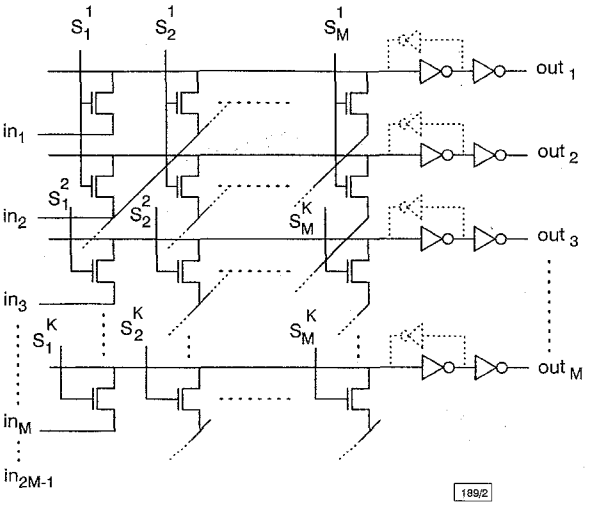

Fig. 2 *Proposed low power barrel shifter with k-way control line splitting*

shifter (where $M = \Sigma^K M^l$). Thus, if the nth bit slice should be shifted out for table lookup and the length of the codeword to be shifted out is N, where $\Sigma^l M^l \le N < \Sigma^{l+1} M^l$ and $1 \le l < K$, we only have to shift out the first $\Sigma^l M^l$ bits by asserting control signals S_n^1, S_n^2, ..., S_n^l and disable other control signals to prevent the other ($M - \Sigma^l M^l$) bits from changing their states and dissipating energy. The circuit overheads of this barrel shifter are several gates to control the gating of control lines and the overhead will not introduce

much extra power consumption if the number of control segments K is not large. Note that the bit lines where all the control transistors are not turned on are now floating and may lose their states due to the leakage current. To prevent this problem, a small feedback inverter may be added to the bit lines of the barrel shifter to make sure these bits remain in their states while all the gating transistors are turned off. However, for a high throughput application like HDTV, those feedback inverters will not really be needed because the operation frequency of the Huffman decoder is usually high.

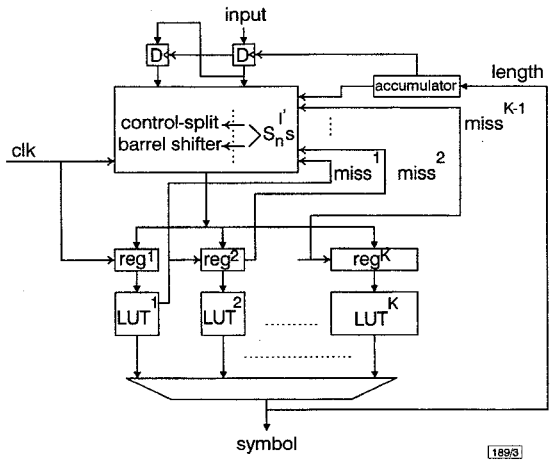

Fig. 3 *Architecture of low power parallel Huffman decoder*

The proposed low power technique can be combined with the non-uniform table partitioning approach [3] to further reduce the power dissipation of the parallel Huffman decoder. Fig. 3 shows the resulting parallel Huffman decoder with K-way control splitting and table partitioning for the barrel shifter and lookup table, respectively. The procedures for this decoder to decode the codeword are described as follows:

Step 1: Start to shift nth bit slice

Step 2: Let $l = 1$

Step 3: The decoder asserts S_n^1, S_n^2, ...S_n^l to shift out first $\Sigma^l M^i$ bits from the M bit slice to LUT^l

Step 4: **If** (the $\Sigma^l M^i$ bit slice matches a codeword in the LUT^l or l == K) **then**

generate the output symbol and the codeword length

Else

assert miss^l, let $l = l + 1$ and goto step 3

Step 5: determine the next bit slice location n and goto step 1

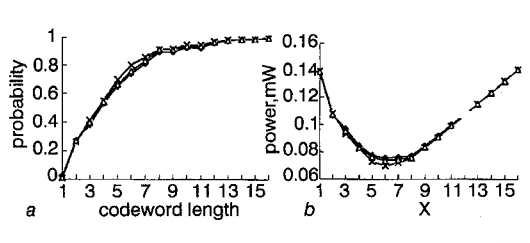

Fig. 4 *Simulation results*

a Cumulative probability of codeword length
b SPICE simulation of power consumption in proposed barrel shifter with two-way control splitting; X: number of bits controlled by S_n^1
—◆— football
—×— table tennis
—△— flower garden

The proposed barrel shifter architecture is simulated using SPICE to verify the power reduction. The inputs to the simulation are real-world codewords extracted from three MPEG-2 sequences: football, table tennis and flower garden. Fig. 4*a* shows the probability of occurrence of the extracted codewords with length $\leq K$ bits, where $1 \leq K \leq 16$. We can see that although the maximum codeword length is 16 bits, ~90 percent of the codewords have a length of less than nine bits. Fig. 4*b* shows the power consumption of the 16 bit barrel shifter with the two-way control line splitting to S_n^1 and S_n^2 ($1 \leq n \leq 16$). The horizontal value X indicates the number of bits controlled by the first control line S_n^1. We can see that the average power of the 16 bit barrel shifter without control splitting (X=16) is ~0.14mW while the average power of the two-way control-split barrel shifter with six bits controlled by S_n^1 (X=6) is ~0.07mW. Therefore, a maximum 50 percent of power saving in the barrel shifter can be achieved with the proposed technique. There will be an even greater power reduction if the probability curve in Fig. 4*a* is sharper.

Conclusion: This Letter presents a technique to reduce power consumption of a parallel Huffman decoder. Because the distribution of the Huffman codeword length is non-uniform and no codeword is the prefix of another codeword, the power dissipation of the barrel shifter can be effectively reduced with low circuit overheads by eliminating the unnecessary bit shifting in the barrel shifter of the parallel Huffman decoder. The proposed technique can be combined with the table partitioning scheme to further reduce the power consumption of the parallel Huffman decoder.

28 October 1997
Electronics Letters Online No: 19980150

Chia-Hsing Lin and Chein-Wei Jen (*Department of Electronics Engineering and Institute of Electronics, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu, 300, Taiwan, Republic of China*)

E-mail: cwjen@twins.ee.nctu.edu.tw

References

1 LEI, S.-M., and SUN, M.-T.: 'An entropy coding system for digital HDTV applications', *IEEE Trans. Circuits Syst. Video Technol.*, 1991, **1**, pp. 147–155

2 CHANG, S.-F., and MESSERSCHMITT, D.G.: 'Designing high-throughput VLC decoder Part I - Concurrent VLSI architectures', *IEEE Trans. Circuits Syst. Video Technol.*, 1992, **2**, (2), pp. 187–196

3 CHO, S.-H., XANTHOPOULOS, T., and CHANDRAKASAN, A.P.: 'Design of low power variable length decoder using fine grain non-uniform table partitioning'. IEEE Int. Symp. Circuits and Systems, 1997, pp. 2156–2159

Performance of least mean absolute third (LMAT) adaptive algorithm in various noise environments

Young Hwan Lee, Jin Dam Mok, Sang Duck Kim and Sung Ho Cho

Convergence properties of the stochastic gradient adaptive algorithm based on the least mean absolute third (LMAT) error criterion are presented. The performance of the algorithm is examined and compared for several different probability densities of the measurement noise in the system identification mode. It is observed that the LMAT algorithm outperforms the LMS algorithm for most noise probability densities, except in the case of exponentially distributed noise.

Introduction: The adaptive least mean square (LMS) algorithm [1] has received a great deal of attention during the last two decades, and is now widely used in variety of applications due to its simplicity and relatively well-behaved performance. The algorithm tries to minimise the mean-squared estimation error at each iteration. More recently, adaptive filtering algorithms that are based on high order error power (HOEP) conditions [2–4] are proposed and their convergence properties have been investigated. Despite their potential advantages, the HOEP adaptive algorithms are much less popular than the LMS algorithm in practice. This seems to be mostly because they can be very sensitive to stability, even though they often show better convergence behaviour.

The paper by Walach and Widrow [2] seems to be the first which deals with the HOEP conditions in stochastic gradient adaptive signal processing. It presented a convergence analysis of the least mean fourth (LMF) algorithm and its family. It was