A One Division per Clock Pipelined Division Architecture Based on LAPR(Lookahead of Partial-Remainder) for Low-power ECC Applications

Hyung-Joon Kwon and Kwyro Lee
Korea Advanced Institute of Science and Technology

Abstract

We propose a pipelined division architecture for low-power ECC applications, which is based on partial-division on group basis and lookahead technique exploiting the linearity in finite field arithmetic. The throughput is one division per clock regardless of the degree of the dividend polynomial. The salient feature of this architecture is that it leads very low power-delay product. To verify the relative performance of the proposed division architecture over the conventional one using LFSR, three RS and BCH code applications were fabricated using 0.8μm double metal CMOS technology. Experimental results show about 32, 65, 67 times improvement in power consumption compared with conventional one using LFSR.

I. Introduction

Division in the finite field GF(2^m) is the most important building block in ECC(Error Correction Coding) systems such as BCH(Bose-Chaudhuri-Hocquenghem) and RS(Reed-Solomon) codes, since these block codings are based on long polynomial divisions[1]. The conventional Euclidean division architecture in finite field uses LFSR(Linear Feedback Shift Register). However, as the high-speed requirement for real-time audio/video coding as well as the low-power requirement for portable applications increase, this serial architecture has shown several limitations as follows. 1)The throughput is limited by the degree of the dividend polynomial. 2)The presence of a global feedback signal imposes severe constraints on the switching speed and necessitates the use of a global clock[2]. 3)This feedback signal limits the degree of parallelism that can be exploited for low-power consumption[3]. 4)The fact that the complete LFSR and serial buffer registers should be clocked for every clock cycle without concerning the change of contents, can not avoid useless power consumption[4]. Therefore, for high-speed/low-power ECC applications, a new division architecture which does not suffer from limitations mentioned above is necessitated.

II. New Division Algorithm based on LAPR

While the conventional division algorithm does symbol or bit basis serial processing, our division algorithm processes on group basis parallel processing. It starts from the definition of $P(x)$ as the long arbitrary dividend polynomial of degree n and $M(x)$ as the fixed divisor polynomial of degree k, i.e., $P(x)=\sum_{i=0}^{n} p_i x^i$ and $M(x)=\sum_{i=0}^{k} m_i x^i$. If we define q as the maximum number that satisfies $n \geq q(k+1)+k$ then the elements in the dividend polynomial can be grouped into q+2 orthogonal groups as follows : $P(x)=\sum_{i=0}^{q} P_i(x)x^{i(k+1)} + P_{-1}(x)$. All of the groups $P_j(x)$ for $q \geq j \geq 0$ has the same format as $S(x)$ defined as $S(x)=(\sum_{i=0}^{k} s_i x^i)x^k$.

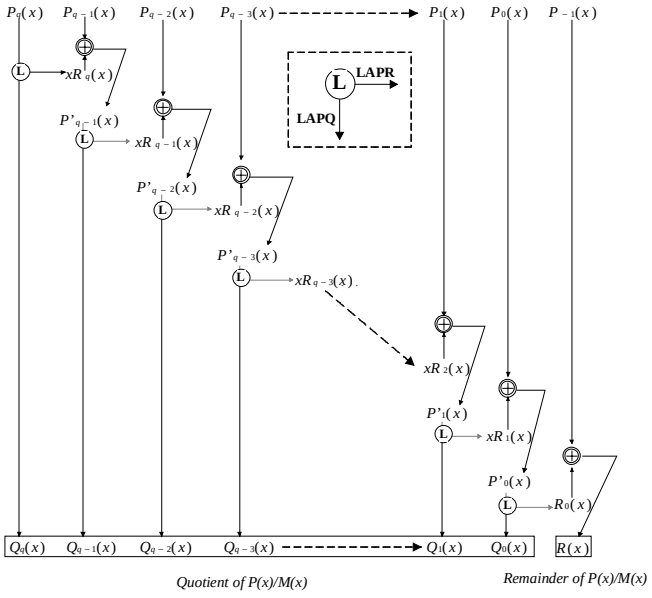


Figure 1 The schematic diagram of new division algorithm based on LAPR for hardware implementation

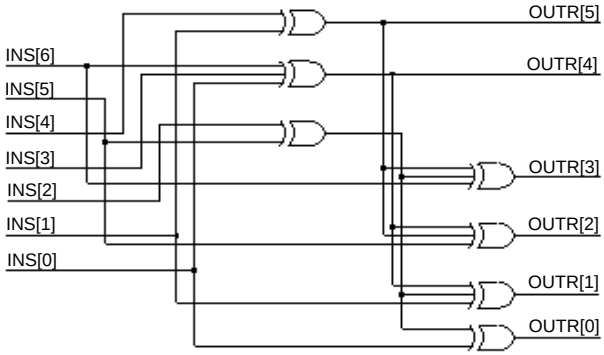
Figure 1 shows our division algorithm schematically for hardware implementation. $Q_q(x)$ and $R_q(x)$ are the quotient and the remainder, respectively, resulting from $P_q(x) / M(x)$. $P'_j(x)$ for $q-1 \geq j \geq 0$ is the sum of 1 left symbol shift of $R_{j+1}(x)$ and $P_j(x)$. Here, to notice is that, in finite field, adding two symbols or polynomials with the same degree does not produce carry, leading to the resulting polynomial $P'_j(x)$ has the same format as $P_j(x)$. All of the $Q_j(x)$ and $R_j(x)$ for $q-1 \geq j \geq 0$ are the quotients and the remainders resulting from $P'_j(x) / M(x)$. We define them as partial-qutient and partail-remainder respectively, since those are the results from a partial-division. The overall quotient of $P(x)/M(x)$ is the weighted sum of all the partial-quotients $Q_j(x)$ for $q \geq j \geq 0$ and the overall remainder is the sum of $R_0(x)$ and last group $P_{-1}(x)$.

Since all of the $P_j(x)$ for $q \geq j \geq 0$ has the same format as $S(x)$, all of the $Q_j(x)$ and $R_j(x)$ can be obtained by looking the results from $S(x) / M(x)$ using identical circuits. To obtain the result from $S(x) / M(x)$ by circuits with less complexity and also by systematic way, we exploited the linearity of the finite field arithmetic[1]. That is $S(x) / M(x)$ is the same as the linear sum of each element in $S(x)$ divided by $M(x)$. For one simple example in the binary field, if the divisor polynomial is $M(x) = x^6 + x^4 + x^2 + x + 1$ then $S(x)$ can be expressed as follows: $S(x) = (\sum_{i=0}^{6} s_i x^i) x^6$. By exploiting the linearity of the finite field arithmetic, all the necessary information to form the lookahead circuits can be listed as shown in Table I.

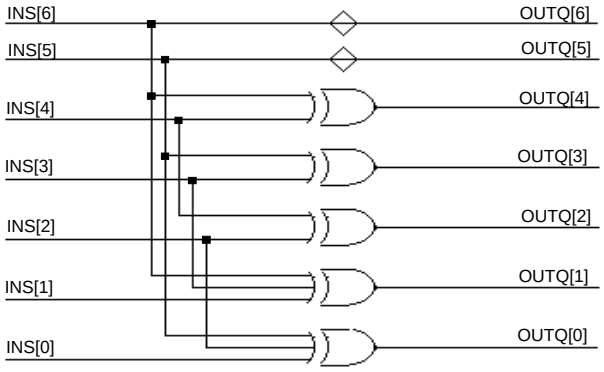
Table I Division table for lookahead circuit :
$M(x) = x^6 + x^4 + x^2 + x + 1$

Input	*S(x)/x*^{6}		
	1000000	1010010	011110
	0**1**00000	0101001	001111
	00**1**0000	0010100	101100
	000**1**000	0001010	010110
	0000**1**00	0000101	001011
	000001**0**	0000010	101110
	000000**1**	0000001	010111
Output of S(x)/M(x)		*Qs(x)*	*Rs(x)*

By superposing the result in Table I we can obtain an optimized lookahead circuit for partial-remainder and partial-quotient as shown in Figure 2. One last thing to notice is, since our algorithm based on LAPR does not need partial-quotients to proceed the division process, partial-quotient lookahead circuitry can be completely eliminated unless application to apply needs quotient explicitly.



a) Lookahead circuit for partial-remainder



b) Lookahead circuit for partial-quotient

Figure 2 Circuit for Lookahead :
$M(x) = x^6 + x^4 + x^2 + x + 1$

III. Division Architecture based on LAPR

Noting the inherent regularity and feedforward natures of our algorithm, we make it fully be pipelined. Figure 3 shows the block diagram of the pipelined architecture based on LAPR. Here, the block FIRST is the register for the first group $P_q(x)$. The q identical blocks INT are intermediate group registers, which form new intermediate groups $P'_j(x)$ for $q-1 \geq j \geq 0$ by adding the partial-remainder from the previous group and the input group $P_j(x)$. The block LAST is the remainder register. Adding the partial-remainder from $P'_0(x)$ and $P_{-1}(x)$ forms the overall remainder. All of the group registers can be implemented using only FFs(Flip-Flop) and EXORs. There are (q+1) identical blocks LOOK-

AHEADR and LOOK-AHEADQ that generate the partial-quotient and partial-remainder respectively. Figure 4 shows the operation diagram of the pipelined architecture. Each group in the dividend polynomial is inserted one by one sequentially to its own specific stage from the first to the last. Each group in the next dividend polynomial can be inserted as soon as the group of the present dividend polynomial of that stage is processed. After (q+2) cycles, all the blocks in Figure 3 operate simultaneously so that the throughput of this pipelined architecture is 1 remainder and 1 quotient per clock cycle. An area efficient sequential architecture based on LAPR is shown in Figure 5. It uses single cell recursively to perform the division process based on LAPR. Every (q+2) cycles, one remainder and one quotient are produced. Although this is slower than the pipelined architecture shown in Figure 3, as far as the authors know, it is still faster than any other division architecture ever reported.

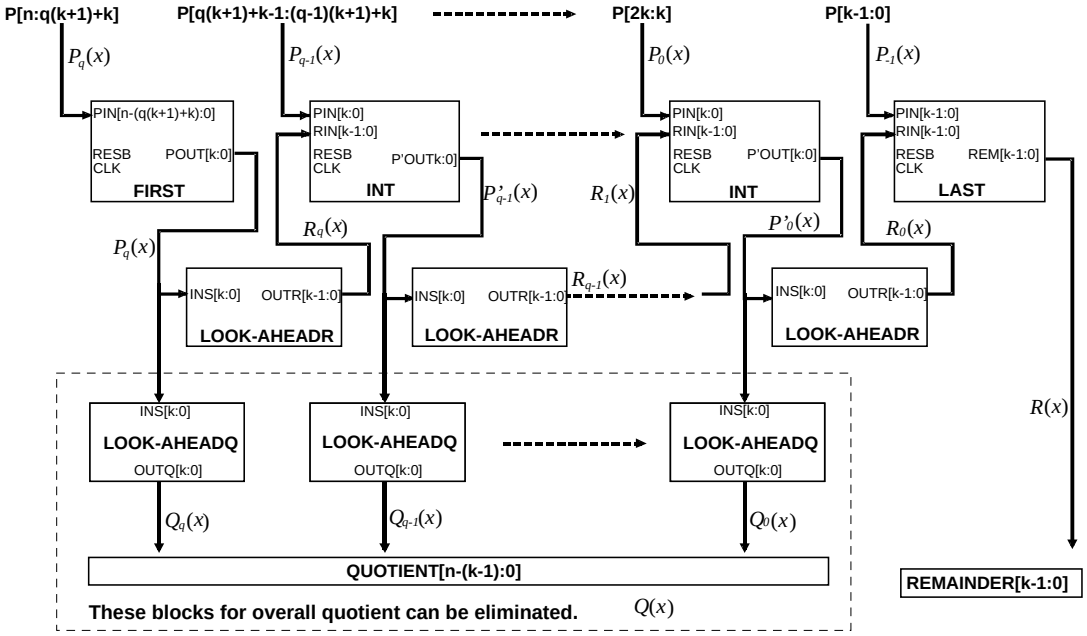


Figure 3 The pipelined division architecture based on LAPR.

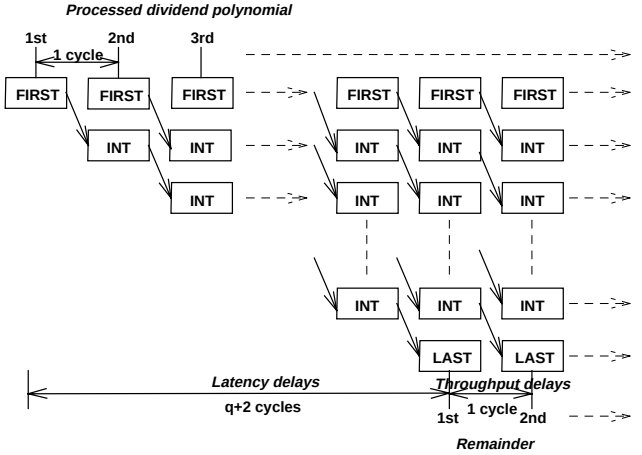


Figure 4 The operation diagram of pipelined division architecture. Here, for simple illustration, we assumed each lookahead circuits are included in its group register block.

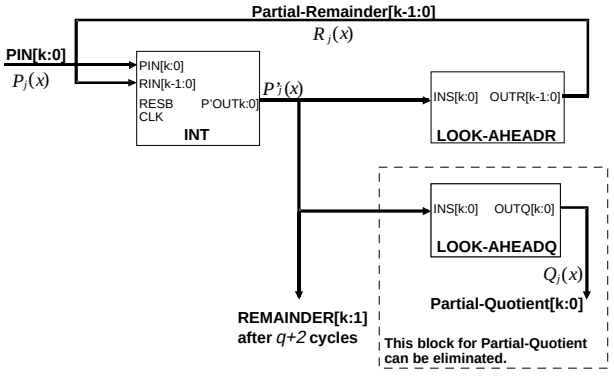


Figure. 5 An area efficient sequential division architecture based on LAPR.

IV. Experimental Verification and Performance Comparisons

To show the superiority of the proposed architecture based on LAPR compared with the

conventional one using LFSR, in terms of speed, area, and power consumption, we designed some popularly used BCH/RS coding applications in COMPASS ASIC development environment using 0.8μm double metal CMOS technology. Three applications: 1) (32,28) RS encoder, 2) (63,51) BCH encoder, 3) syndrome generator for (63,51) BCH decoder, were designed as benchmark circuits to verify the relative performance of the proposed division architecture over the conventional LFSR one. The (32,28) RS code in GF(2^m) and the (63,51) BCH code are now being used in CD(Compact Disk) error correction coding[6] and AMPS(Advanced Mobile Phone Service) cellular phone respectively. The chip micro-photograph is shown in Figure 6.

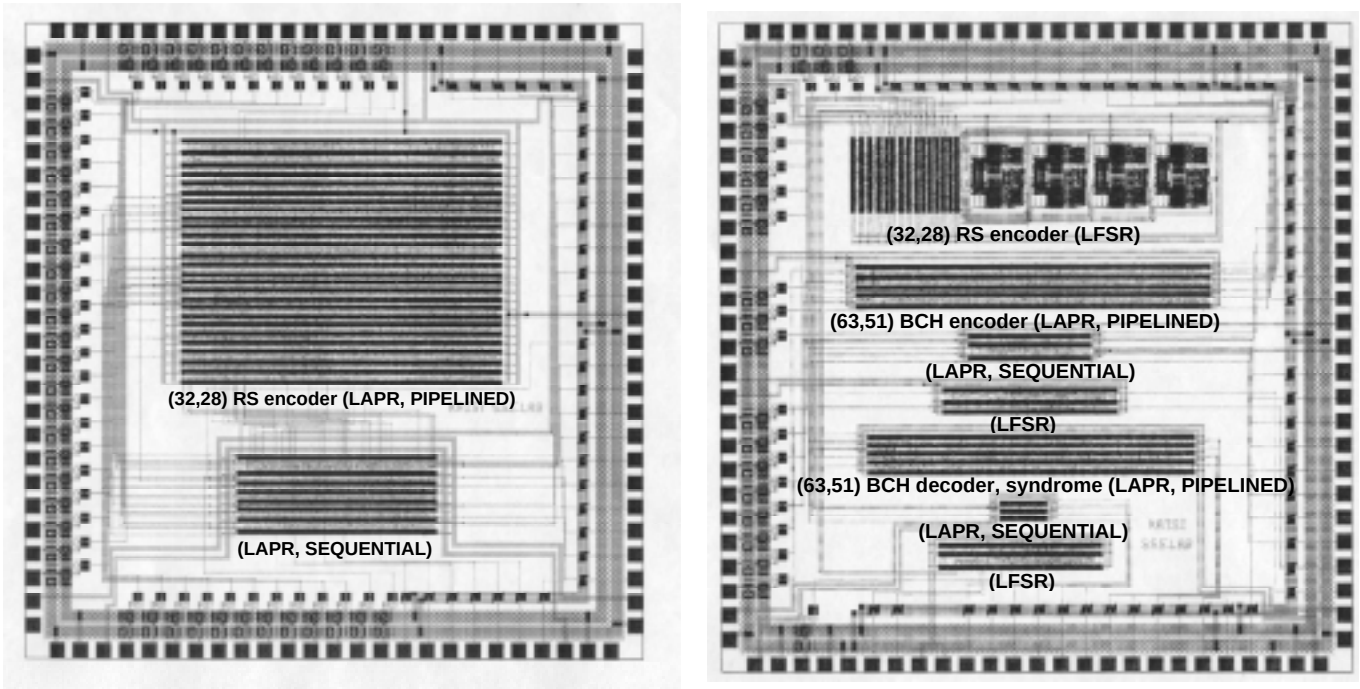


Figure 6 Photo micrograph of the fabricated chips

The experimental results are summarized in Table II. The clock frequency used to obtain the same throughput(500K div/sec) is shown in the second column. Power consumption with supply voltage 5V, is measured and listed in the 4th column. It indicates that the pipelined architectures based on LAPR show 17, 28, 29 times improvement in power consumption compared with those using LFSR. The corresponding improvement for sequential architectures based on LAPR are 10, 13, 18 respectively. To show the power reduction that can be obtained by the architecture driven voltage scaling, we measured power consumption at the minimum supply voltage at which circuits are in proper operation. Since reducing the supply voltage comes at the cost of increased gate delays, as the used clock speeds are higher, lower functional throughput is inevitable. The 5th column in Table II shows this minimum power consumption. It indicates that further power reduction can be obtained by voltage scaling. Pipelined architectures based on LAPR show 32, 65, 67 times improvement in power consumption compared with those using LFSR. The corresponding improvements for sequential architectures based on LAPR are 14, 22, 28 respectively. To show the power efficiency in terms of energy aspect, the normalized power-delay product is depicted on Figure 7. All the circuits are in operation at 5V supply voltage and 10MHz clock frequency. It indicates that pipelined and sequential architecture based on LAPR has very small power-delay product compared with conventional one using LFSR. We also can see, at identical clock frequency, fully-pipelined LAPR architecture produce orders of magnitude big boost in speed for very little power cost.

Table II Summary of experimental results for identical throughput

Architecture (Throughput 500K division/sec)	Clock frequency	Latency delays (cycle)	Power Consumption @VDD=5V	Minimum Power consumption	Multipliers used	Size (mm)	Number of Transistor
1) (32,28) RS code => Error correction code of Compact Disk[5], encoder Divisor polynomial : $M(x) = (x+\alpha^0)(x+\alpha^1)(x+\alpha^2)(x+\alpha^3)$ Degree of dividend : 31, The finite field used : GF(2^8)							
Pipelined(LAPR)	**0.5M**	**7**	**4.186mW**	**1.051mW@2.5V**	**Not used**	**2.50 x 1.89**	**34881**
Sequential(LAPR)	**3.5M**	**7**	**7.201mW**	**2.422mW@2.9V**	**Not used**	**1.56 x 0.85**	**6702**
Serial(LFSR)	16M	32	72mW	33.29mW@3.4V	ROM	3.35 x 0.78	4207 + 4x(2^8 x 8) ROM
2) (63,51) BCH code => Error correction code of AMPS, encoder Divisor polynomial : $M(x) = x^{12}+x^{10}+x^8+x^5+x^4+x^3+1$ Degree of dividend : 62, The finite field used : GF(2)							
Pipelined(LAPR)	**0.5M**	**6**	**0.557mW**	**0.118mW@2.3V**	**-**	**2.71 x 0.35**	**6852**
Sequential(LAPR)	**3M**	**6**	**1.210mW**	**0.352mW@2.7V**	**-**	**0.94 x 0.21**	**1455**
Serial(LFSR)	31.5M	63	15.81mW	7.747mW@3.5V	-	1.34 x 0.21	2018
3) (63,51) BCH code, syndrome generator as a decoder building block Divisor polynomial : $M(x) = x^6+x^4+x^2+x+1$ Degree of dividend : 62, The finite field used : GF(2)							
Pipelined(LAPR)	**0.5M**	**11**	**0.509mW**	**0.107mW@2.3V**	**-**	**2.50 x 0.31**	**6085**
Sequential(LAPR)	**5.5M**	**11**	**0.815mW**	**0.255mW@2.8V**	**-**	**0.36 x 0.16**	**526**
Serial(LFSR)	31.5M	63	14.81mW	7.258mW@3.5V	-	1.25 x 0.21	1958

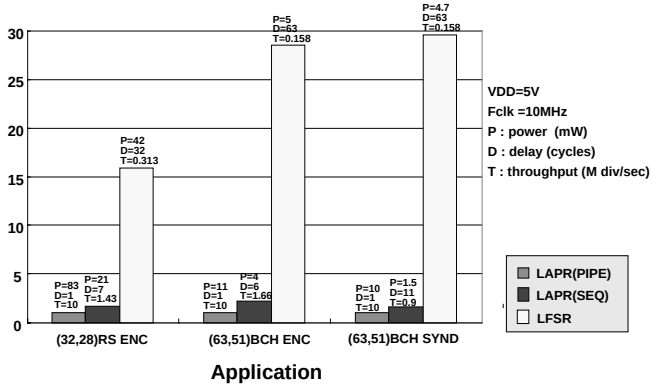


Figure 7 Normalized power-delay product

V. Conclusion

We proposed long polynomial division architectures based on LAPR division algorithm. Both the partial-division on group basis and lookahead technique exploiting the linearity of the finite field arithmetic, enables complete elimination of polynomial multiplication leading to highly increased throughput per unit time. Experimental verification for three benchmark circuits show that at identical throughput, pipelined architecture based on LAPR consumes about 32, 65, 67 times smaller power compared with conventional one using LFSR. Since proposed division algorithm based on LAPR is efficient, regular and easily expandable, it can be used directly in VLSI implementation of various ECC applications where high-speed and/or low-power is dictated for application to communication, optical disks, portable equipment and computer systems.

Acknowledgment

We would like to thank LG Semiconductor and IDEC for the fabrication of chips. We also grateful to Samsung Electronics for their Humantech support.

References

[1] Richard E. Blahut, *Theory and Practice of Error Control Codes*. Addison-Wesley, 1984.

[2] Gadiel Seroussi, "A Systolic Reed-Solomon Encoder," *IEEE trans. Inform. Theory*, vol.37, no. 6, pp. 1217-1220, July. 1991.

[3] A.P. Chandrakasen., et al., "Low-power Chipset for Portable Multimedia Applications," in the proceedings of the '*94 IEEE International Solid State Circuits Conference*, Feb. 1994

[4] Menahem Lowy, "Low power spread spectrum code generator based on parallel shift register implementation," in proceeding of the *'94 IEEE Symposium on Low Power Electronics.*, pp. 22-23, Aug. 1994.

[5] Osamu K., el al, "A High Speed Signal Processing for Quadruple Speed CD-ROM," *IEEE trans. Consumer Electronics*, vol. 40, no. 3, pp679-685, Aug. 1994.