 ISSN: 2321-8169
224 - 228

Study of Bit-Serial Multiplier in Finite Fields GF (2^m)

Riddhish Shukla[1]: Mtech. VLSI Design, VIT University, Tamil Nadu, India.
rids.shukla31@gmail.com

Kulin Shah[1]: Mtech. VLSI Design, VIT University, Tamil Nadu, India.
coolinshah@gmail.com

Raj Chaurasia[1]: Mtech. VLSI Design, VIT University, Tamil Nadu, India.
rajrohaniit@gmail.com

Sivanatham S[2] : Associate Professor, VLSI Division, VIT University, Tamil Nadu, India.
ssivanatham@vit.ac.in

***Abstract*—**In this paper, polynomial based Bit-Serial Multiplier is studied. MSB first bit multiplication has followed for different operand length. This multiplier can be used for order less than what it is designed called versatility. Here gated clock technique is used for reducing the power consumption. Row of tri-state buffer is used for decreasing the critical path delay.Performance of this multiplier is compared with others, based on criteria space complexity,clock latency.Given architecture provides better performance in terms of low power and high speed.

Keywords*—Galois field,Polynomialmultiplication,Low power design,Bit-serial multiplier.*

*****

I. Introduction:

Finite field manipulation is important tool in various applications like error correcting codes, computer algebra and cryptography.Security is the main aim in cryptography which mainly depends upon complexity of mathematical problem. Finite field has main application in cryptography and in cryptography[4]ECC(elliptic curve cryptosystem) is becoming more popular due to its advantages. Main operation in ECC is point multiplication and performance of any system can be evaluated based on how fast it can do point multiplication[12]. If design of multiplier is done efficiently then performance of our cryptosystem will be enhanced [1].

Finite field contains finite number of elements and we can performbasic operation like addition,subtraction,multiplicationand division on it.Finite field of order 2^m is our Galois field which is special interest because they are particularly efficient for implementation in hardware. Finite field of order 2^m denotes that there is m bit input which has order m-1 polynomial representation.

Finite field which has only two element can be defined as Galois field and denoted by GF(2^m).It has an attractive feature of carry free arithmetic and another advantage is that it is available in different equivalent representation of field elements like polynomial, normal or dual basis.GF(2^m) multiplier has three major categories bit-serial,digit-serial and bit-parallel multipliers[4].

Bit serial multiplier has O (m) area requirement, on the other hand bit parallel multiplier has O(m^2) area requirement. Besidesthat serial multiplier needs m clock cycle or move where parallel multiplier needs only one clock cycle so there is trade-off between area requirement and clock cycle[8].

For fixed field size multiplier drawback is that if size of operand is changed then multiplier needs to be re-designed. Here irreducible polynomial is used to achieve versatility. If input has polynomial of order m then irreducible polynomial has order m+1.Actually in cryptography we encrypt our message and for different word there is variable length code so to determine that each and every time we have to re-design the multiplier if we use fixed length multiplier but here by the usage of irreducible polynomial redesigning eliminated.

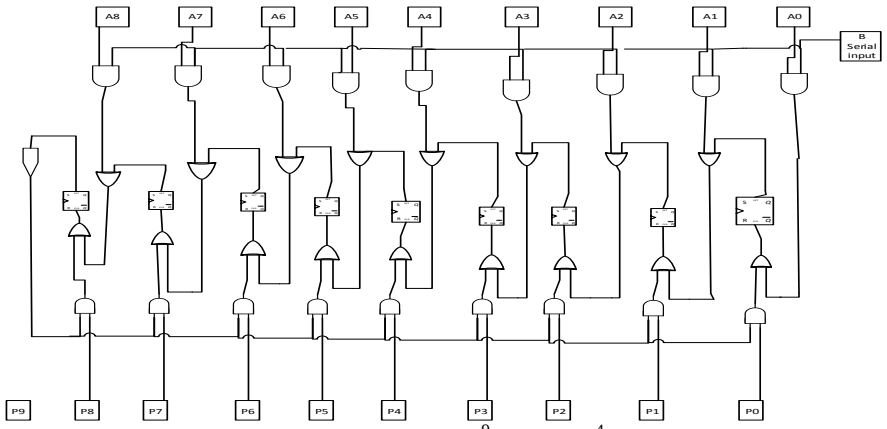

Fig 1. Problem of utilization GF (2^9)as GF (2^4) Multiplier [13]

 ISSN: 2321-8169
224 - 228

Now a day's power play very important role in any architecture means designed architecture should consume less power. Most of the power dissipation occurs due to switching activity so if we reduce switching at unwanted flip-flop then we can reduce power consumption. For that we have used gated clock technique. In this technique clock will not be provided to the flip-flop which are not in the use. We have also used rows of tri state buffer for lesser critical path delay. In other multiplier model the feedback is given from the MSB only so we have wait until the carry is forwarded to MSB. Whereas here we do not have to wait until MSB, we get the feedback after every operation.

II. Architecture and its operation

In GF (2^m) any element can be stated as binary polynomial of at most m-1. Addition operation is same as subtraction, both are carry free [4], but multiplication is different. Multiplication is complex therefore it has become subject of research now a day because by any means if multiplication process for any system becomes simpler then system performs is enhanced. In 1989 Itoh and Tsuji [7] designed multiplier which had low complexity after that evolution in design of multiplier started. After so much evolution multiplier with less critical path delay and lesser power is designed and we have studied that architecture in detail in this paper and also explained how this multiplier is efficient.

The major problem in multipliers which are designed previously (Fig.1) is that they are not versatile means they cannot be used for operand length other than what they are designed for. But this multiplier can perform multiplication for variable length operand. Based on which bit enter first multiplier can be of two types LSB first multiplier and MSB first multiplier. In this architecture MSB first

Technique [2] is used. Here example is presented with GF(2^9) multiplier which is used as GF(2^4) multiplier. For that two inputs and one irreducible polynomial has taken. Actually in GF(2^m) when degree of result is more than m-1, it needs to be reduced for that irreducible polynomial is used. It is implemented with bit shift and XOR. Generally, an irreducible polynomial of degree m could be presented as

$$P(x) = x^m + x^k + \sum_{j=1}^{k-1} a_j x^j + 1$$

where $a_j \in GF(2)$ and $1 < k < m$[7]

That means for polynomial of degree m degree of irreducible polynomial is m+1 and the mandatory condition for irreducibility is that the MSB, LSB and also at least one of middle bits m+1 bit string be one. A and B input taken randomly and P is taken according to given condition.

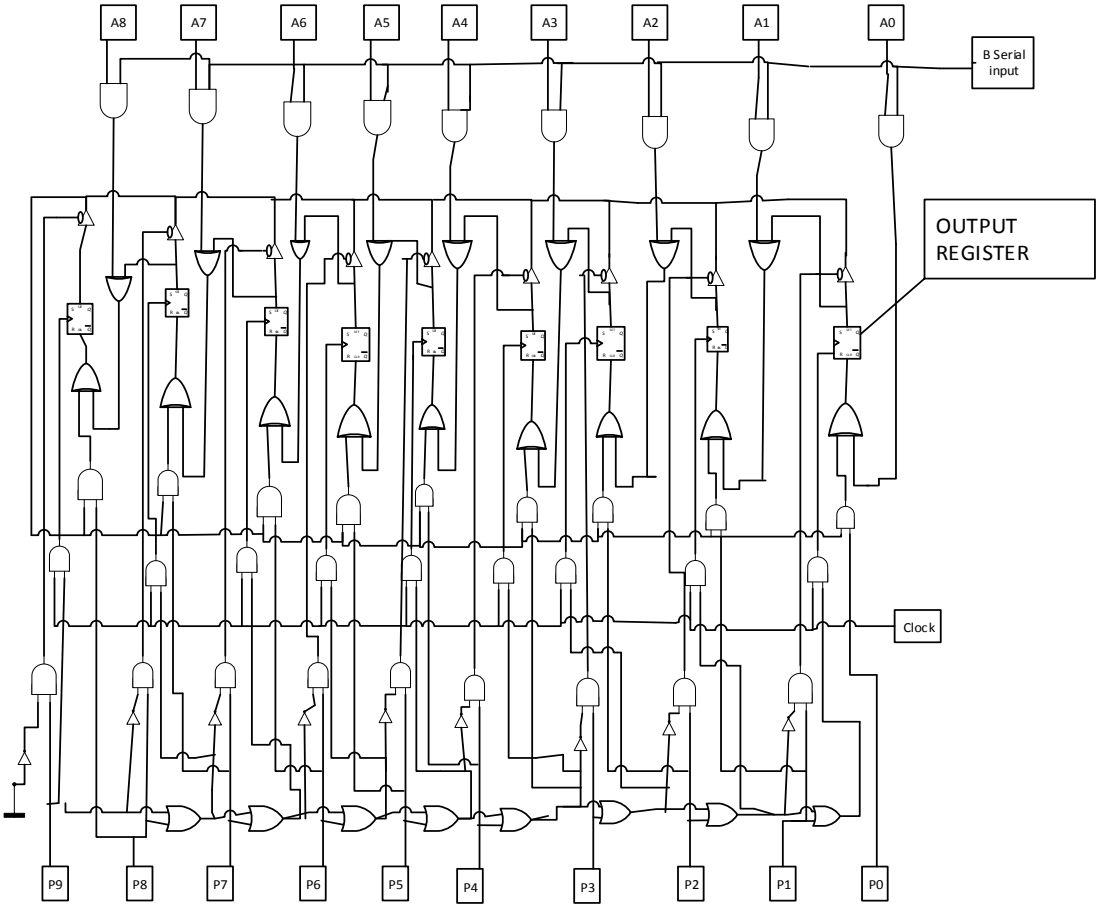

Fig 2. Architecture of Multiplier of GF(2^9) to multiply operands[13]

 ISSN: 2321-8169
224 - 228

$$A(X) = X^3 + X^1 + 1$$

$$B(X) = X^2 + X$$

$$P(X) = X^4 + X^2 + 1$$

As shown in Figure(2) between bits of P registers OR operation is done initially so only for first clock cycle its computed then it always available for other clock cycle for given operand length if operand length changes then it will compute for that particular cycle after that it will available for all clock cycle. In this way by using serial structure of OR gates critical path is reduced considerably [2]. But number of control signal has increased compared to previously designed structures. The power consumption is reduced because of gated clock technique is used. Control signals are generated such a way that they should on only those flip-flops which are required to be on during our operation it will turn off flip-flops which are not used in operation. In this way power consumption is reduced.

Here feedback is given via tri state buffer and because of we don't want to wait until carry is generated from MSB therefore feedback is given from every bit and not from MSB[2]. As GF (2^m) field has only two possible coefficient (1 or 0) in its polynomial presentation therefore here taken example can be written in binary form as A[1011] B[0110] and P[10101].

In Galois field multiplication operation is done as A×B mode P. so according to that first multiplication of A and B happens so by multiplying 4 bit by 4 bit we get 8 bit that are 00111010 but output should not be more than m-1 degree therefore irreducible polynomial helps us to reduce that degree by XOR operation. If the bits are more even that then of irreducible polynomial then zero padding is done in that polynomial then XOR operation is performed so after that degree will be reduced, still degree is higher than m-1 then previous operation is performed and try to compress it in m-1 degree. So according to that 111010 XOR with 101010 is done and the answer is 10000 further XOR operation is done with 10101 and the answer is 0101 [11]. Threshold between 0's and 1's in the internal control register is determined by degree of irreducible polynomial. Due to one of tristate buffer is enabling, feedback is given through proper bit of s register. (Fig. 3) Period of clock cycle is determined by delay of critical path. Delay of critical path is determined by summation of delay of tristate buffer, 2-input XOR gate and 2-input AND gate.

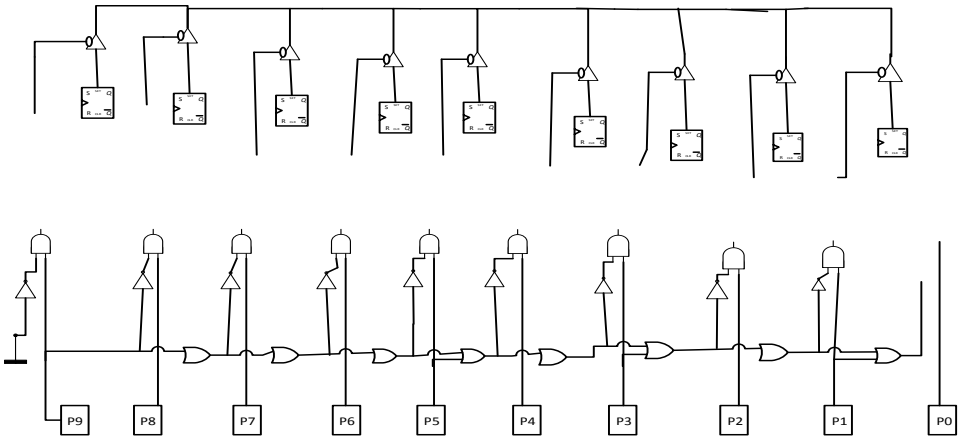

Fig 3. To Configure the Tri-State Buffer from Polynomial Register[13]

III. Results and discussion

The Serial Bit multiplier is designed and simulated in Model sim and synthesized using Quartus Tool. The Multiplier is implemented on FPGA DE2-115 Board. And its structure is compared with other structure as shown in Table 1.

Multiplication Implementation	In this architecture	Wallece tree multiplier	Dadda multiplier
AND	16	38	36
XOR	8	22	20
REG	13	12	12
OR	4	7	8
Tristate buffer	4	-	-

Critical path delay	$T_{AND}+T_{XOR}+T_{Tri\text{-}state\ buffer}$	-	-

Table 1. Comparison With different Structures for Multiplication

From this table one can inferred that number of gate count used in given architecture is less than that used in wallece tree multiplier and dadda multiplier leading to less critical path delay. Critical path delay mainly depends upon number of gates that comes from input to output in worst case.

Here critical path delay is got from timing analysis is 1.437ns and power consumed is 562.32mW.
Here it is clearly seen from table that the multiplier which are not used in the Galois field has large number gates are used. So compared to that, architecture which has been studied has lesser numberof gates. Time for one GF (2^m) multiplication

$$(2T_{AND} + T_{XOR} + T_{NOT} + (m+1)T_{OR})$$

Summary of structure: M states > no of bit in the input here are 4.

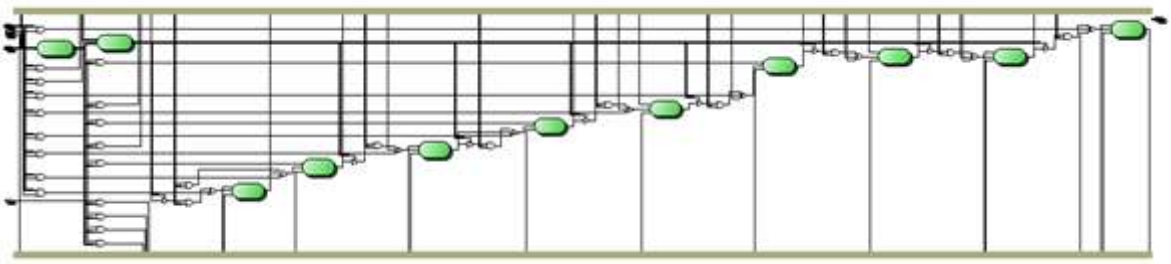

Fig 4.RTL in Quartus Tool

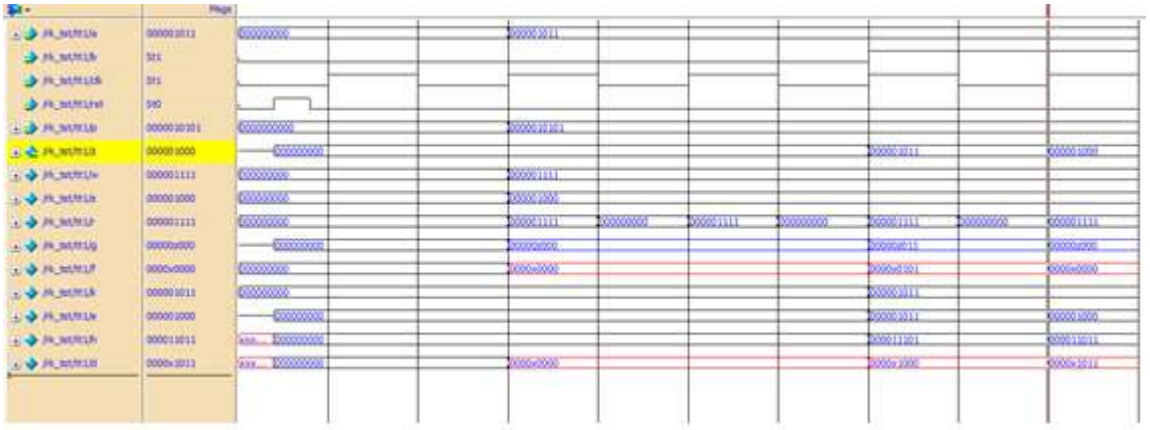

Fig 5.Simulation Result in Model Sim

Here A is loaded with 1011 and B is serially given according to that different wire output is shown and output is highlighted.

Conclusion

A flexible architecture for a polynomial basis multiplier over GF (2^m) is studied. Series of tri-state buffers and a set of control signals are withincreated to achieve a versatile property together with a lower power dissipation property. The key advantages of the architecture are (1) its compatibility with arbitrary Galois Field size (2) its hardware easiness which results in lesser area

Implementation, (3) low power consumption by using the gated clock method(4) enhancement of maximum clock frequency due to the decreasing of critical path delay. Results have definite the appropriateness of design in cryptography applications when there are low hardware properties and power limitations.

References

[1] Sudhakar, M., Ramachandruni Venkata Kamala, and M. B. Srinivas. "A Unified, Reconfigurable Architecture for Montgomery Multiplication in Finite Fields GF (p) and GF (2^n)." VLSI Design, 2007. Held jointly with 6th International Conference on Embedded Systems., 20th International Conference on. IEEE, 2007.

[2] Kitsos, Paris, George Theodoridis, and O. Koufopavlou. "An efficient reconfigurable multiplier architecture for Galois field GF (2 m)."Microelectronics Journal 34.10 (2003): 975-980.
[3] Selimis, G. N., Fournaris, A. P., Michail, H. E., & Koufopavlou, O. (2009). Improved throughput bit-serial multiplier for GF (2 m) fields. Integration, the VLSI journal, 42(2), 217-226.
[4] A.P.Fournaris,O.Koufopavlou,Versatilemultiplierarchitecturesin GF (2k) fields usingtheMontgomeryMultiplicationAlgorithm,INTEGRATION,the VLSI Journal41 (3)(2008)371–384.(paged).
[5] Nikooghadam, Morteza, Ehsan Malekian, and Ali Zakerolhosseini. "A versatile reconfigurable bit-serial multiplier architecture in finite fields GF (2m)." Advances in Computer Science and Engineering. Springer Berlin Heidelberg, 2008. 227-234.M.A.Garcı´a-Martı´nez, R.Posada-Go´ mez, G.M.Luna,FPGAimplementationof an efficientmultiplieroverfinitefields GF (2m), in:IEEEProceedingsofthe 2005 InternationalConferenceonReconfigurableComputingandFPGAs, 2005, pp.68–82.
[6] Garcia-Martinez, M.A., Posada-Gomez, R., Morales-Luna, G. and Rodriguez-Henriquez, F., 2005, September. FPGA implementation of an efficient multiplier over finite fields GF (2/sup m/). In Reconfigurable Computing and FPGAs, 2005. ReConFig 2005. International Conference on (pp. 5-pp). IEEE.
[7] Itoh, Toshiya, and Shigeo Tsujii. "Structure of parallel multipliers for a class of fields GF (2 m)." Information and computation 83.1 (1989): 21-40.
[8] Li, Hua, and Chang N. Zhang. "Efficient cellular automata based versatile multiplier for GF (2^ m)." J. Inf. Sci. Eng. 18.4 (2002): 479-488.
[9] Mishra, Ravi Shankar, Puran Gour, and Mohd Abdullah. "Design & Implementation of 8 Bit Galois Encoder for on FPGA Secure Data Transmission."
[10] Lakhandra kumar, Dr k l sudha. Implementation of Galois field arithmetic unit on FPGA.(IJIRCCE Vol 2 Issue 6 June 2014)
[11] James e Steen. Digital Computer arithmetic DATAP.
[12] Behrouz forouzan.CryptographyNetwork Security.
[13] Zakerolhosseini, Ali, and Morteza Nikooghadam. "Low-power and high-speed design of a versatile bit-serial multiplier in finite fields GF (2 m)."INTEGRATION, the VLSI journal 46.2 (2013): 211-217.