IMPLEMENTATION OF FIXED AND FLOATING POINT
DIVISION USING DHVAJANKA SUTRA

R.THAMIL CHELVAN¹, S.ROOBINI PRIYA²
¹Assistant Professor, ²Student Scholars, Department of Electronics and Communication Engineering,
Sree Sastha Institute of Engg and Tech
¹tamilchelvan@yahoo.com, ²ruby.sce25@gmail.com

ABSTRACT
This paper proposes the implementation of RSA encryption/ decryption algorithm using the algorithms of
Ancient Indian Vedic mathematics that has been modified to improve the performance. The recently proposed
divider architecture is used in RSA circuitry for division operation. The most significant aspect of this paper is
the development of division architecture based on straight division algorithm of ancient Vedic mathematics and
embedding it in RSA encryption/decryption circuitry for improved efficiency. The modification that has been
attained is done in the addition circuitry of the division architecture. The coding is done in Verilog HDL and
FPGA synthesis is done using Xilinx Spartan library. The results show that RSA circuitry implemented using
Vedic division is efficient in terms of area and speed, compared to its implementation using conventional
division architecture.

Index terms: Vedic math’s, Different Division Architecture, Dhvajanka sutra, RSA algorithm.

1.INTRODUCTION
MATHEMATICS is a mother of all sciences. Mathematics is full of magic and mysteries. The ancient Indians
were able to understand these mysteries and develop simple keys to solve these mysteries. Thousands of years
ago the Indians used these techniques in different fields like construction of temples, astrology, medicine,
science etc., due to which INDIA emerged as the richest country in the world. The Indians called this system of
calculations as “THE VEDIC MATHEMATICS”. Vedic Mathematics is much simpler and easy to understand
than conventional mathematics.

The ancient system of Vedic Mathematics was re-introduced to the world by Swami Bharati Krishna Tirthaji
Maharaj, Shankaracharya of Goverdhan Peath. “Vedic Mathematics” was the name given by him. Bharati
Krishna, who was himself a scholar of Sanskrit, Mathematics, History and Philosophy, was able to reconstruct
the mathematics of the Vedas. Vedic Mathematics introduces the wonderful applications to Arithmetrical
computations, theory of numbers, compound multiplications, algebraic operations, factorizations, simple
quadratic and higher order equations, simultaneous quadratic equations, partial fractions, calculus, squaring,
cubing, square root, cube root, coordinate geometry and wonderful Vedic Numerical code. Conventional
mathematics is an integral part of engineering education since most engineering system designs are based on
various mathematical approaches. All the leading manufacturers of microprocessors have developed their
architectures to be suitable for conventional binary arithmetic methods. The need for faster processing speed is
continuously driving major improvements in processor technologies, as well as the search for new algorithms.
A Divider is one of the key hardware blocks in most of applications such as digital signal processing [2],
encryption and decryption algorithms in cryptography [3] and in other logical computations [4]. With advances
in technology, many researchers have tried to design multipliers which offer either of the following- high speed,
low power consumption, regularity of layout and hence less area or even combination of them in multiplier.
The Vedic multiplier is considered here to satisfy our requirements.

In this work, we present the different division architectures, in the second section we present Division based
on Dhvajanka sutra, third section deals about RSA algorithm and the last section deals about results and
conclusion.

2. THE DIFFERENT DIVISION ARCHITECTURES
2.1. SRT DIVISION (Sweeney, Robertson and Tocher)
Digit recurrence algorithms use subtractive methods to calculate quotients one digit per iteration. SRT division
is the name of the most common form of digit recurrence division algorithm. The input operands are assumed to
be represented in a normalized floating point format with n bit significant in sign-and-magnitude representation.
The algorithms presented here are applied only to the magnitudes of the significant of the input operands.
Techniques for computing the resulting exponent and sign are straightforward.

2.2. NEWTON RAPHISON METHOD
The continuous Newton method for the solution of nonlinear systems of equations arises from Newton’s method
when damping with infinitesimal step sizes is applied. To numerical analysts this method is of theoretical
interest because the topological behavior of the continuous dynamical system corresponding to a Newton vector
field is better understood than the discrete dynamical systems arising from Newton’s method with finite damping.

2.3. LONG DIVISION METHOD

Just as multiplication of counting numbers is based on repeated additions, the inverse operation of division may be understood in terms of repeated subtractions. A useful way to visualize $73 \div 6$ is to think of $73$ objects divided equally among 6 people. When each person receives one object, the number of objects is reduced by 6. If each person receives 5 objects, the initial number 73 is reduced by $5 \times 6 = 30$. The process ends when the number of objects remaining is less than 6. In the end, each person will receive 12 objects, with one remaining and this can be represented by $73 = 6 \times 12 + 1$. The division problem $73 \div 6$ can be posed as the search for whole numbers $q$ and $r$ for which $73 = 6 \times q + r$ where $0 < r < 6$. More generally, the division problem $a \div b$ is the search for whole numbers $q$ and $r$ for Which $a = b \times q + r$, where $0 < r < b$. An efficient way of doing successive subtractions and solving for the numbers $q$ and $r$ is the long division algorithm.

3. THE DHVAJANKA SUTRA

Shift Register. Zero padding is needed to represent a number to higher number of bits. For example if a four bit number “0111” is represented in eight bit format then the representation becomes “00000111”. Here zero padding has been executed based on the requirement. Bidirectional shift register performs the operation of shifting both to the left or right depending upon the control signal ‘carry’ from the second subtractor. The contents of the Quotient Array Register are the input to the Bidirectional Shift register. The Divider architecture combining the exponent extraction and the ‘Dhvajanka’ Sutra is shown in Fig.-8. The basic building blocks of the architecture are (i) Exponent Extraction Unit, (ii) Divider using ‘Dhvajanka’ sutra, (iii) Subtractor, (iv) Quotient Array Register and (v) Bidirectional Shift Register. Zero padding is needed to represent a number to higher number of bits. For example if a four bit number “0111” is represented in eight bit format then the representation becomes “00000111”. Here zero padding has been executed based on the requirement. Bidirectional shift register performs the operation of shifting both to the left or right depending upon the control signal ‘carry’ from the second subtractor. If carry=0 then the input bits are shifted to the left and if carry=1 then input bits are shifted to the right. The result of the second subtractor is fed to the shifter to control number shifting. The contents of the Quotient Array Register are the input to the Bidirectional Shift register.

4. RSA ALGORITHM

RSA algorithm can generally be further classified into key generation algorithm, encryption algorithm, and decryption algorithm.

The RSA key generation algorithm can be described in the following five steps.
1. Generate two large random and distinct primes $P$ and $Q$
2. Calculate $N = P \times Q$
3. Choose a random integer $J$, $1 < J < K$, such that $\text{gcd} (J, K) = 1$
4. Compute the unique integer $I$, $1 < I < K$, Such that $J^I \equiv 1 \text{ (mod K)}$
5. Public key is $(N, J)$ and private key is $(N, I)$

While RSA encryption algorithm is described by $L = M^J \text{ mod N}$.

![Divider architecture using Dhvajanka sutra](image-url)
The decryption algorithm is described by

\[ M = L^U \mod N, \]

Where \( L \) represents the cipher text and \( M \) represents the message.

The standard techniques for providing privacy and security in data networks include encryption/decryption algorithms such as Advanced Encryption System (AES) (private-key) and RSA (public-key). RSA is one of the safest standard algorithms, based on public-key, for providing security in networks. While hardware implementation of this algorithm tends to be faster compared to its software counterpart, there is a scope for further improvement of Performance of RSA hardware. One of the most time consuming processes in RSA encryption/decryption algorithm is the computation of \( a^b \mod n \) where \( a \) is the text, \( (b,n) \) is the key and this paper examines how this computation could be speeded up drawing up on the Indian Vedic Mathematics.

5. VERIFICATION AND IMPLEMENTATION

Division involving a 12 bit divider and 16 bit divider is implemented using VLSI and it is proved that the division is implemented with less time computation, minimized area and reduced delay. Further, Vedic division is applied to RSA algorithm and the division involved in RSA algorithm has been simplified with the application of vedic mathematics in it.

RESULTS AND CONCLUSION

The various division architectures are compared with the Vedic division architecture, and it has been proved that the Vedic division can be implemented with reduced time, minimized area and reduced delay. The RSA encryption/decryption circuitry achieves a significant improvement in performance using the Vedic Divider. It is found that when implemented with the Vedic division algorithm, the RSA circuitry has less timing delay compared to its implementation using traditional multipliers and division algorithms. It has already been demonstrated that Vedic hierarchical dividers are efficient than conventional dividers in terms of area/speed. For the Xilinx Spartan family, it is found that the gate delay for RSA circuitry using 8x8 overlay multiplier architecture and 16 bit by 16 bit Vedic division is 1.507 \( \mu \)s with area of 15235 (14942 FMAP and 293 HMAP). While the gate delay of RSA circuitry using 8x8 overlay multiplier and restore division is 2.838 \( \mu \)s with area of 14141 (14077 FMAP, 64 HMAP) that of non-restore division with 8x8 overlay multiplier is 2.828 \( \mu \)s with area of 6689 (6616, 73 HMAP). Hence, we can dynamically reduce the power consumption in the divider circuit using the Vedic mathematics.

Inference: If 5678 is divided by 34
Quotient :167
Remainder :0

Inference: If 8976 is divided by 67
Quotient :140
Remainder :16
Output Using RSA algorithm:

Key Generation:

Power Calculation:

REFERENCES


