International Journal of Research Studies in Computer Science and Engineering (IJRSCSE)
Volume 2, Issue 7, July 2015, PP 18-20
ISSN 2349-4840 (Print) & ISSN 2349-4859 (Online)
www.arcjournals.org

Bound Estimate of Bitwise Exclusive OR Operation

WANG Xingbo

Department of Mechatronics
Foshan University
Foshan City, PRC
wxbmail@msn.com

Abstract: *Bitwise exclusive OR operation has been widely applied in electronic engineering, automatic engineering, mechatronic engineering, computer science and cryptography. Bound estimate of the operation becomes necessary for algorithm designs of many areas. This article presents bound estimates of both a general form and 3 incremental forms of the operation. The general form can be used in analysis of the operation for general purpose and the 3 forms can be used in incremental analysis of the operation. Theorems with their proofs are presented in detail.*

Keywords: *Bitwise exclusive OR operation, Bound, Estimate*

1. INTRODUCTION

Bitwise exclusive OR operation, which is denoted by symbol ^ in computer *C* language, is a very important operation in computer science as seen in bibliographies [1] and [2]. It performs the operation according to the rules that 0^0=0, 0^1=1, 1^0=1 and 1^1=0. The article [3] has proved that the ^ operation does not fit the distributive law over addition, that is to say that $a \wedge (b+c)$ is not necessarily equal to $a \wedge b + a \wedge c$. This results in difficulty in analysis of the operation in abstract way because the analysis cannot rely on the traditional way of thinking.

As the ^ operation has played important roles in more and more applications, e.g. the many analytic formulas in article [4], estimate of the operation or bound estimate of the operation becomes necessary so as for us to design proper algorithms. This paper makes an investigation on the issue and presents the results.

2. PRELIMINARIES

We need the following lemmas for later deductions.

Lemma 1 ([5][6][7]). The floor function $\lfloor x \rfloor$ is an integer such that $x-1 < \lfloor x \rfloor \le x$ and it holds that, for any real *x* and *y*, $x \le y$ yields $\lfloor x \rfloor \le \lfloor y \rfloor$ and $x \ge y$ yields $\lfloor x \rfloor \ge \lfloor y \rfloor$, and for any integer *n* and real *x*, $\lfloor n + x \rfloor = n + \lfloor x \rfloor$.

Lemma 2 ([7]). Total valid bits of positive integer α's binary representation is $\lfloor \log_2 \alpha \rfloor + 1$.

3. MAIN RESULTS AND PROOFS

We obtain estimates of ^ operation that calculates $\alpha \wedge (\alpha+\delta), \alpha \wedge (\alpha-\delta), (\alpha+\delta) \wedge (\alpha-\delta)$ and $\alpha \wedge \beta$.

Theorem 1. Let δ and α be positive integers that satisfy $\alpha \ge \delta$, then it holds

$$\alpha \wedge (\alpha+\delta) \le 4\alpha - 1, \alpha \wedge (\alpha-\delta) \le 2\alpha - 1, (\alpha-\delta) \wedge (\alpha+\delta) \le 4\alpha - 1$$

Proof. Let $\alpha = (\alpha_{n-1}\alpha_{n-2}\alpha_{n-3}...\alpha_0)_2$ and $\delta = (\delta_{n-1}\delta_{n-2}...\delta_0)_2$ be the *n*-bits binary representations of α and δ respectively, then the binary representation of $\alpha+\delta$ contains most *n*+1 binary bits and that of $\alpha-\delta$ contains most *n* binary bits, namely

$$\alpha+\delta=(\alpha_{n-1}\alpha_{n-2}\alpha_{n-3}...\alpha_0)_2+(\delta_{n-1}\delta_{n-2}...\delta_0)_2=(\chi_n\chi_{n-1}\chi_{n-2}\chi_{n-3}\cdots\chi_0)_2\ ,$$
$$\alpha-\delta=(\alpha_{n-1}\alpha_{n-2}\alpha_{n-3}...\alpha_0)_2-(\delta_{n-1}\delta_{n-2}...\delta_0)_2=(\eta_{n-1}\eta_{n-2}\eta_{n-3}...\eta_0)_2$$

where $\chi_i(i=0,...,n),\eta_i(i=0,...,n-1)$ are number 0 or 1.

Hence it yields

$$\alpha\wedge(\alpha+\delta)=(\alpha_{n-1}\alpha_{n-2}\alpha_{n-3}...\alpha_0)\wedge(\chi_n\chi_{n-1}\chi_{n-2}\chi_{n-3}\cdots\chi_0)_2=(\theta_n\theta_{n-1}...\theta_0)_2\le(\underbrace{1...1}_{n+1})_2=2^{n+1}-1$$
$$\alpha\wedge(\alpha-\delta)=(\alpha_{n-1}\alpha_{n-2}\alpha_{n-3}...\alpha_0)_2\wedge(\eta_{n-1}\eta_{n-2}\eta_{n-3}...\eta_0)_2=(\vartheta_{n-1}\vartheta_{n-2}...\vartheta_0)_2\le(\underbrace{1...1}_{n})_2=2^n-1$$
$$(\alpha+\delta)\wedge(\alpha-\delta)=(\chi_n\chi_{n-1}\chi_{n-2}\cdots\chi_0)_2\wedge(\eta_{n-1}\eta_{n-2}...\eta_0)_2=(\omega_n\omega_{n-1}...\omega_0)_2\le2^{n+1}-1$$

where $\theta_i(i=0,1,...,n),\vartheta_j(j=0,1,...,n-1)$ and $\omega_k(k=0,1,...,n)$ are respectively binary number 0 or 1.

By Lemma 1 and 2, it immediately leads to

$$\alpha\wedge(\alpha+\delta)\le2^{\lfloor\log_2\alpha\rfloor+2}-1\le2^{2+\log_2\alpha}-1=4\alpha-1$$
$$\alpha\wedge(\alpha-\delta)\le2^{\lfloor\log_2\alpha\rfloor+1}-1\le2^{1+\log_2\alpha}-1=2\alpha-1$$
$$(\alpha+\delta)\wedge(\alpha-\delta)\le2^{n+1}-1\le4\alpha-1$$

By Theorem 1 and Lemma 2, the following Corollary 1 is easy to derive out.

Corollary 1. For arbitrary two positive integers α and β, it holds

$$\alpha\wedge\beta\le\min(4\alpha-1,2\max(\alpha,\beta)-1)$$

Proof. By Lemma 2, total valid bits in α's and β's binary representations are respectively $\lfloor\log_2\alpha\rfloor+1$ and $\lfloor\log_2\beta\rfloor+1$. Hence the twos binary representations must be

$$\alpha=(0...01\underbrace{\alpha_k...\alpha_2\alpha_1}_{k})_2\ \text{ where }\ k=\lfloor\log_2\alpha\rfloor$$
$$\beta=(0...01\underbrace{\beta_l...\beta_2\beta_1}_{l})_2\ \text{ where }\ l=\lfloor\log_2\beta\rfloor$$

Without loss of generality, we assume such that $\alpha\le\beta$. Then it yields

$$\alpha\wedge\beta=(0..001\underbrace{\chi_l\cdots\chi_2\chi_1}_{l})_2\le2^{l+1}-1=2^{\lfloor\log_2\beta\rfloor+1}-1\le2\beta-1$$

By theorem 1, it immediately results in

$$\alpha\wedge\beta\le\min(4\alpha-1,2\beta-1)$$

Corollary 1 gives a general estimate for $\alpha\wedge\beta$. The following theorem 2 gives estimate for $\alpha\wedge\beta$ in case of $2^{k-1}\le\alpha,\beta\le2^k-1$

Theorem 2. Let positive integer *k*, α and β satisfy $2^{k-1}\le\alpha\le2^k-1$ and $2^{k-1}\le\beta\le2^k-1$; then $\alpha\wedge\beta\le2^{k-1}-1$.

Proof. First we can see that any integer j that fits $2^{k-1}\le j\le2^k-1$ can be expressed by $j=2^{k-1}+\delta$ where $0\le\delta<2^{k-1}$, and j's binary representation is $j=(0...01\underbrace{\delta_{k-1}\cdots\delta_2\delta_1}_{k-1})_2$, where $\delta_i(1\le i\le k-1)$ is binary bit 0 or 1.Therefore, without loss of generality, we assume

$$\alpha=(0...01\underbrace{\alpha_{k-1}...\alpha_2\alpha_1}_{k-1})_2\ \text{ and }\ \beta=(0...01\underbrace{\beta_{k-1}...\beta_2\beta_1}_{k-1})_2$$

where $\alpha_i,\beta_i(1\le i\le k-1)$ is 0 or 1.

Then it leads to

$$\alpha \wedge \beta = (0..00\underbrace{\chi_{k-1}\cdots\chi_2\chi_1}_{k-1})_2$$

where $\chi_i = \alpha_i \wedge \beta_i (1 \le i \le k-1)$ is 0 or 1.

Hence it holds

$$\alpha \wedge \beta \le (0...00\underbrace{1\cdots 11}_{k-1})_2 = 2^{k-1} - 1$$

4. Conclusion

The wide applications of bitwise exclusive OR operation in electronic engineering, computer engineering, automatic engineering, mechatronic engineering and cryptography have already demonstrated its universal importance in front of us. Consequently, algorithm design related to the operation has been concentrated in the related areas. Bound estimate for an operation, as is known, is the base and prerequisite for algorithm design, especially for analysis of an algorithm. Unfortunately, few articles have been found in the topic. In this article, we first obtain estimate of $\alpha \wedge (\alpha+\delta), \alpha \wedge (\alpha-\delta)$ and $(\alpha+\delta) \wedge (\alpha-\delta)$ because the expressions $\alpha+\delta$ and $\alpha-\delta$ are frequently used incremental analysis. Since $\alpha \wedge \beta$ is a general ^ operation, we also obtain its estimate for general purpose. I hope more articles to disclose properties of bitwise operations can be seen in the future.

Acknowledgements

The research work is supported by the national Ministry of science and technology under project 2013GA780052, Department of Guangdong Science and Technology under projects 2012B011300068, Foshan Bureau of Science and Technology under projects 2013AG10007, Special Innovative Projects 2014KTSCX156 from Guangdong Education Department, and Chancheng government under projects 2013B1018 and 2013A1021. The authors sincerely present thanks to them all.

References

[1]. Wikipedia. Bitwise operation. https://en.wikipedia.org/wiki/Bitwise_operation. Aug., (2010).

[2]. Stephen Prata, C Primer Plus(Fifth Edition), US:Sams,2004,chapter 15

[3]. Wang Xingbo. Analysis on Operation Law of Bitwise Operation with a Proof for a Modulo Identity, Journal of Foshan University (Natural Science Edition),29(3),53(2011).

[4]. WANG Xingbo. Analytic Formulas for Computing LCA and Path in Complete Binary Trees[J].International Journal of Scientific and Innovative Mathematical Research (IJSIMR),3(4),81(2015)

[5]. Graham R. L., Knuth D. E., Patashnik O., Concrete Mathematics: A Foundation for Computer Science, Addison-Wesley, 1994,ch3,pp67-101.

[6]. WANG Xingbo, A Mean-value Formula for the Floor Functions on Integers, Mathproblems,2(4),136(2012).

[7]. WANG Xingbo, Some Supplemental Properties with Appendix Application of Floor Function, Journal of Science of Teacher's College and University (In Chinese), 34(3),7(2014)

Author's Biography

WANG Xingbo, was born in Hubei, China. He got his Master and Doctor's degree at National University of Defense Technology of China and had been a staff in charge of researching and developing CAD/CAM/NC technologies in the university. Since 2010, he has been a professor in Foshan University, still in charge of researching and developing CAD/CAM/NC technologies. Wang has published 8 books, over 70 papers and obtained more than 20 patents in mechanical engineering.