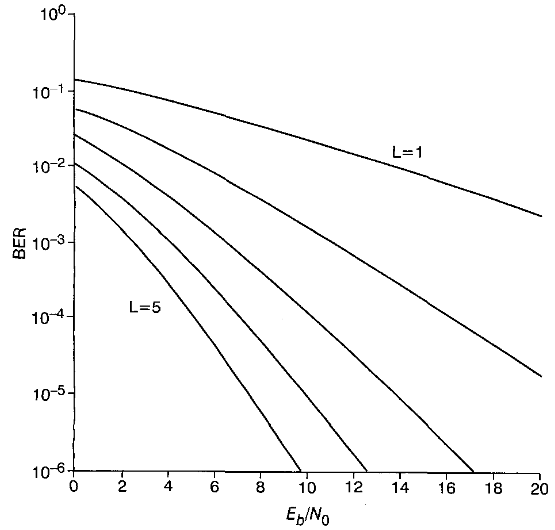

Fig. 1 *BER for BPSK in Nakagami fading ($L = 5, 4, 3, 2, 1, m = 1$)*

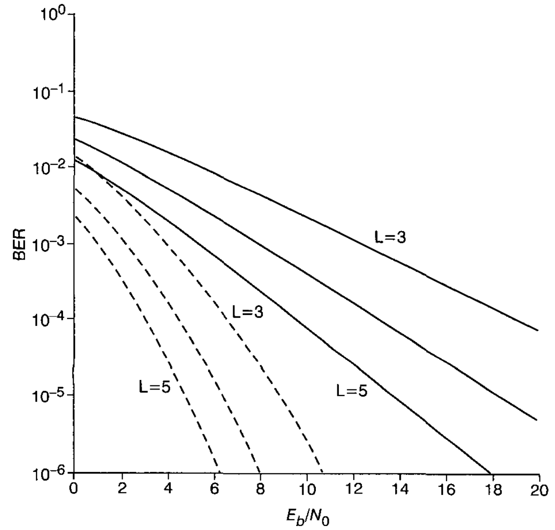

Fig. 2 *BER for BPSK in Nakagami fading ($L = 5, 4, 3$; $m = 2$ and $m = 0.5$)*

Since branch fading is assumed to be statistically independent, and $a_{k,l}$s are Nakagami random variables, γ also follows the Nakagami distribution with parameter a Lm [6], with the pdf

$$p(\gamma) = \frac{m^{Lm}}{\Gamma(Lm)} \gamma^{Lm-1} e^{-m\gamma} \tag{3}$$

Results: The error probability performance can be obtained by averaging the probability of error conditioned on the fading over the pdf of (3). For BPSK the unconditional probability of error can be expressed thus

$$P_b = \frac{1}{2} \int_0^{\infty} erfc(\sqrt{\gamma d}) \frac{m^{Lm}}{\Gamma(Lm)} \gamma^{Lm-1} e^{-m\gamma} d\gamma \tag{4}$$

Now, define $\delta = d/(m(1+d))$ and $\beta = \gamma m(1+d)$, where $d = E_s/N_0$, then the error probability can be written as

$$P_b = \frac{1}{2(1+d)^{Lm}} \int_0^{\infty} erfc(\sqrt{\beta\delta}) \frac{\beta^{Lm-1}}{\Gamma(Lm)} e^{-\beta(1-m\delta)} d\beta \tag{5}$$

The integral part can be evaluated which can thus be expressed in the form

$$P_b = \frac{(\delta(1+d))^{-Lm}}{2\sqrt{\pi}} \frac{\Gamma(Lm+0.5)}{\Gamma(Lm+1)} \times {}_2F_1\left(Lm, Lm + \frac{1}{2}, Lm + 1; m - \frac{1}{\delta}\right) \tag{6}$$

where ${}_2F_1(\cdot, \cdot, \cdot; \cdot)$ is the Gaussian confluent hypergeometric function. Let us define $x = m - (1/\delta) = -m/d$, we can express (6) with the help of [7] as

$$P_b = \frac{\delta^{-Lm}}{2\sqrt{\pi}(1+d)^{Lm}} \frac{\Gamma(Lm+0.5)}{\Gamma(Lm+1)} (1-x)^{-Lm} \times {}_2F_1\left(Lm, \frac{1}{2}, Lm+1; \frac{x}{x-1}\right) \tag{7}$$

Also using [7], we can express ${}_2F_1(Lm, 1/2, Lm+1; x/(x-1)) = Lm(x/(x-1))^{-Lm} B_{x/(x-1)}(Lm, 0.5)$ Finally, putting this equation in (7), we have the strikingly simple result

$$P_b = \frac{1}{2\sqrt{\pi}} \frac{\Gamma[Lm+0.5]}{\Gamma[Lm]} B_{x/(x-1)}(Lm, 0.5) \tag{8}$$

where, $B_x(\cdot, \cdot)$ is the incomplete beta function defined in [7].

The expression in (8) is plotted in Figs 1. and 2 for selected diversity branches L and Nakagami parameter m. Fig. 1 shows for $L = 5, 4, 3, 2, 1$ and $m = 1$ (Rayleigh). Fig. 2 shows for $L = 5, 4, 3$ and $m = 0.5$ and 2. We have checked the results with the literature and perfect agreement exists [4].

Conclusion: A simple but exact expression has been derived for BPSK with an MRC receiver in Nakagami fading. The expression is valid for all values of m, fast to compute and is computationally efficient.

Acknowledgment: The authors acknowledge the support of KFUPM.

22 April 2002
Electronics Letters Online No: 20020997
DOI: 10.1049/el:20020997

A.B. Adinoyi and S.A. Al-Semari (*Electrical Engineering Department, King Fahd University of Petroleum & Minerals, KFUPM Box 207, Dhahran 31261, Saudi Arabia*)

E-mail: adinoyi@kfupm.edu.sa

References

1 SCHWARTZ, M., BENNETTE, W.R., and STEIN, S.: 'Communication systems and techniques' (McGraw-Hill, 1966)

2 NAKAGAMI, M.: 'The m-distribution – a general formula of intensity distribution of rapid fading in statistical methods in radio wave propagation' *in* HOFFMAN, W.G. (Ed.): 'Statistical Methods in Radio Wave Propagation' (Pergamon, Oxford, England, 1960)

3 BRAUN, R., and DERSCH, U.: 'A physical mobile radio channel model', *IEEE Trans. Veh. Technol.*, 1991, **VT-40**, pp. 472–482

4 AALO, V., and PATTARAMALI, S.: 'Average error rate for coherent MPSK signals in Nakagami fading channel', *Electron. Lett.*, 1996, **32**, (17), pp. 1538–1539

5 ANNAMALI, A.: 'Analysis of selection diversity on Nakagami fading channels', *Electron. Lett.*, 1997, **33**, (7), pp. 548–549

6 AL-HUSSAINI, E., and AL-BASSIOUNNI, A.: 'Performance of MRC diversity systems for the detection of signals in Nakagami fading', *IEEE Trans. Commun.*, 1985, **33**, pp. 1315–1319

7 GRADSHTEYN, I.S., and RYZHIK, I.M.: 'Table of integrals, series, and products' (Academic Press, San diego, CA, 1984)

Improved Yen–Joye's authenticated multiple-key agreement protocol

Min-Shiang Hwang, Chih-Wei Lin and Cheng-Chi Lee

An authenticated multiple-key agreement protocol is proposed. The protocol is not only secure against the unknown-key attack but also more efficient than other protocols.

Introduction: Diffie and Hellman first proposed key agreement protocol to establish a session key for two parties [1]. However, the protocol was later proven to be vulnerable to the unknown-key attack by Diffie *et al.* [2] because the protocol did not include any key authentication process during the negotiation between the two parties [3, 4].

In 1997, Harn first proposed the authenticated key agreement protocol [5] without using a one-way hash function [6]. In 1998, Harn and Lin proposed an authenticated multiple-key agreement protocol based on the Diffie–Hellman distribution scheme [7]. There are two main features in this protocol: it operates without using a one-way hash function and it enables two communication entities to share multiple secret keys.

Later, Yen–Joye [8] indicated that the Harn–Lin protocol is not secure because an attacker can successfully forge a short-term public key pair and pass the verification equation. Then, they proposed an improved Harn–Lin protocol to get rid of this shortcoming. However, in 1999, Wu *et al.* [9] pointed out that the Yen–Joye protocol is insecure and can be successfully attacked the same way as the Harn–Lin protocol. Wu *et al.* then proposed a protocol to enhance the security. Nevertheless, the protocol violated the original expectation of the Harn–Lin protocol that no one-way hash function should be used in the authenticated key agreement protocol.

In this Letter, we shall propose a modification of the Yen–Joye protocol. The modification does not only ameliorate the security but also is more efficient than Harn's protocol proposed in 2001 [10].

Review of Yen–Joye protocol: In this Section, we shall briefly review the Yen–Joye protocol [8]. There are two phases in the protocol. The first phase is the authentication phase where two users exchange n temporary random public keys in an authenticated way. The second phase is the key-sharing phase where the users share $n^2 - 1$ secret keys with each other.

There are two users Alice and Bob who want to establish multiple keys by the protocol. Here, we only describe what Alice has to do because Bob has to do basically the same. Initially, the system has a large prime p, and α is a primitive number in $GF(p)$. Alice has a long-term secret key x_A and the corresponding long-term public key $y_A = \alpha^{x_A} \bmod p$. Then Alice randomly generates two short-term secret keys k_{A1} and k_{A2} and computes their corresponding short-term public keys $r_{A1} = \alpha^{k_{A1}} \bmod p$ and $r_{A2} = \alpha^{k_{A2}} \bmod p$, respectively. The range of r_{A1} and r_{A2} is set to be $(\lceil p/2, p-1 \rceil)$ so that no attacker can forge the keys. Alice computes the signature s_A through r_{A1} and r_{A2} as

$$s_A = x_A - (r_{A1} \cdot r_{A2}) \cdot k_A \bmod (p-1) \qquad (1)$$

where $k_A = k_{A1} \cdot k_{A2} \bmod p$. Finally, Alice sends r_{A1}, r_{A2}, s_A, $cert(y_A)$ to Bob, where $cert(y_A)$ is a certificate for Alice's public key y_A. After receiving them, Bob verifies them via the computation as follows:

$$y_A \equiv ?(r_{A1} \cdot r_{A2})^{(r_{A1} \cdot r_{A2})} \alpha^{s_A} \bmod p \qquad (2)$$

If it holds, Bob establishes the multiple secret keys in the second phase. Bob can derive the session keys as follows:

$$\begin{cases} K_1 = r_{A1}^{k_{B1}} \bmod p \\ K_2 = r_{A2}^{k_{B1}} \bmod p \\ K_3 = r_{A1}^{k_{B2}} \bmod p \\ K_4 = r_{A2}^{k_{B2}} \bmod p \end{cases}$$

Here, three of the four keys can be used because of perfect forward secrecy [11]. Thus, three authenticated session keys can be established in this protocol.

Improved protocol: The Yen–Joye protocol is an improvement on the Harn–Lin protocol. However, according to Wu *et al.*, the Yen–Joye protocol is no more secure than its predecessor. They pointed out the Yen–Joye protocol cannot resist the same attack that bothers the Harn–Lin protocol. The attacker can forge a pair $\{r'_{A1}, r'_{A2}\}$ in the range $(\lceil p/2, p-1 \rceil)$ to satisfy $r'_{A1} r'_{A2} = r_{A1} r_{A2}$ at the probability of greater than 1/18. Although Wu *et al.* later proposed an enhanced protocol with a one-way hash function, this improved protocol violates the original expectation from the Harn–Lin protocol that no one-way hash function should be used in the authenticated multiple keys agreement protocol. In 2001, the Harn–Lin proposed an improved authenticated multiple keys agreement protocol and claimed their protocol can eliminate the attack from [8, 9].

However, the Harn–Lin protocol is not as efficient as the Yen–Joye protocol. In this Letter, we propose two straightforward modifications to enhance the security of the Yen–Joye protocol. The proposed protocol can withstand the attack on Wu *et al*'s scheme and is more efficient than [10]. First, we suggest that the pair of short-term public keys r_{A1} and r_{A2} in the generation phase should be prime numbers. This modification can help the new scheme prevent the attacker from forging another pair (r'_{A1}, r'_{A2}) because the prime numbers are unique. In addition, it obeys the original requirement of the Harn–Lin protocol that the range of r_{A1} and r_{A2} should be in $(1, p-1)$. Secondly, we suggest that the great common divisor (GCD) of r_{A1} and r_{A2} should be equal to 1. This suggestion is to prevent the attacker from finding the factor q of r_{A1} or r_{A2}. Furthermore, the range of r_{A1} and r_{A2} will fall in the $(\lceil p/2 \rceil, p-1)$ as the Yen–Joye protocol proposed. Both of the modifications of on the Yen–Joye protocol can make it secure against any forgery of the pair (r_{A1}, r_{A2}). Besides, our modification protocol is more efficient than the Harn–Lin protocol [10] because we only perform the exponentiation computation four times, less than six times required by the Harn–Lin protocol.

Conclusion: We have proposed an improved scheme to enhance the security of the Yen–Joye protocol. We require that r_{A1} and r_{A2} should be primes or GCD(r_{A1}, r_{A2}) should be equal to 1 to withstand the attack of forging another pair (r'_{A1}, r'_{A2}) so that $r_{A1} \cdot r_{A2} = r'_{A1} \cdot r'_{A2}$. The pair r_{A1} and r_{A2} should be made unique so that no attacker can find another pair to replace them. Furthermore, the Harn–Lin protocol [10] uses six exponentiation computations, while the Yen–Joye scheme four takes only. That means the Harn–Lin protocol is less efficient.

In this Letter, we have proposed two straightforward modifications to withstand the forgery attack on the Yen–Joye protocol. The proposed protocol retains the original expectation on the Harn–Lin protocol that the range of the short-term public key be $(1, p-1)$. Furthermore, the new protocol uses fewer exponentiation computations than the Harn–Lin protocol [10].

Acknowledgment: This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract no.: NSC90-2213-E-324-004.

8 April 2002

Electronics Letters Online No: 20020998
DOI: 10.1049/el:20020998

Min-Shiang Hwang and Chih-Wei Lin (*Institute of Networks and Communications, Chaoyang University of Technology, 168 Gifeng E. Rd., Wufeng, Taichung County, Taiwan 413, R.O.C.*)

E-mail: mshwang@cyut.edu.tw

Cheng-Chi Lee (*Department of Computer and Information Science, National Chiao-Tung University, 1001 Ta Hsueh Road, Hsinchu, Taiwan, R.O.C.*)

References

1 WHITFIELD DIFFIE, and HELLMAN, M.: 'New directions in cryptology', *IEEE Trans. Inf. Theory*, 1976, **IT-22**, (6), pp. 644–654

2 WHITFIELD DIFFIE, VAN OORSCHOT, P.C., and WIENER, M.J.: 'Authentication and authenticated key exchanges', *Des., Codes Cryptogr.*, 1992, **2**, (2), pp. 107–125

3 CHENG-CHI LEE, MIN-SHIANG HWANG, LI-HUA LI: 'A new key authentication scheme based on discrete logarithms', *Appl. Math. Comput. (to be published)*

4 ERIC JUI-LIN LU, and MIN-SHIANG HWANG: 'An improvement of a simple authenticated key agreement algorithm', *Pak. J. Appl. Sci.*, 2002, **2**, (1), pp. 64–65

5 LEIN HARN: 'Digital signatures for Diffie-Hellman public keys without using one-way function', *Electron. Lett.*, 1997, **33**, (2), pp. 125–126

6 MIN-SHIANG HWANG, CHIN-CHEN CHANG, and KUO-FENG HWANG: 'A watermarking technique based on one-way hash functions', *IEEE Trans. Consum. Electron.*, 1999, **45**, (2), pp. 286–294

7 LEIN HARN, and HUNG-YU LIN: 'An authenticated key agreement protocol without using one-way functions'. Proceedings of the 8th National Conference on Information Security, Kaohsiung, Taiwan, May 1998 pp. 155–160

8 SUNG-MING YEN, and JOYE, M.: 'Improved authenticated multiple-key agreement protocol', *Electron. Lett.*, 1998, **34**, (18), pp. 1738–1739

9 TZONG- SUN, WEI-HUA HE, and CHIEN-LUNG HSU: 'Security of authenticated multiple-key', *Electron. Lett.*, 1999, **35**, (5), pp. 391–392

10 HARN, LEIN, and LIN, HUNG-YU: 'Authenticated key agreement without using one-way hash functions', *Electron. Lett.*, 2001, **37**, (10), pp. 629–630

11 LIN, C.H., and LEE, P.J.: 'Security of interactive DSA batch verification', *Electron. Lett.*, 1994, **30**, (19), pp. 1592–1593

Key function of normal basis multipliers in *GF*(2^n)

Haining Fan and Yiqi Dai

A new definition of the key function in $GF(2^n)$ is given. Based on this definition, a method to speed up software implementations of the normal basis multiplication is presented. It is also shown that the normal basis with maximum complexity can be used to design low complexity multipliers. In particular, it is shown that the circuit complexity of a type I optimal normal basis multiplier can be further reduced.

Introduction: An important advance in $GF(2^n)$ arithmetic is the Massey-Omura algorithm. It is well known that the realisation of $GF(2^n)$ operations can be made more efficient by choosing optimal normal basis or low complexity normal basis [1]. Since the complexity of the normal basis multipliers depends on the choice of key function for multiplication, it is desirable to have a key function with minimal complexity to implement the multiplication algorithm [2].

In this Letter, we give a new definition of the key function and present a method to speed up software implementations of the normal basis multiplication. We also show that the circuit complexity of a type I optimal normal basis multiplier can be further reduced.

Preliminaries: Let γ be an element of $GF(2^n)$, for simplicity, denote γ^{2^i} by γ_i. Given a normal basis $N=\{\beta_0, \beta_1, \beta_2, \ldots, \beta_{n-1}\}$ of $GF(2^n)$ over $GF(2)$, a field element A can be represented by a binary vector $(a_0, a_1, \ldots, a_{n-1})$ with respect to this basis as $A=\sum_{i=0}^{n-1} a_i \cdot \beta_i$, where $a_i \in GF(2)$ and $i=0, 1, \ldots, n-1$.

For $1 \le i \le n-1$, let $\beta_0\beta_i=\sum_{j=0}^{n-1}\phi_{i,j}\beta_j$ be the expansion of $\beta_0\beta_i$ with respect to the normal basis N, $\phi_{i,j} \in GF(2)$. Let $R=\{0,1,\ldots,n-1\}$, $S_i=\{j|\phi_{i,j}=1\}$, $h_i=|S_i|$, and $T_i=\{j|\phi_{i,j}=0\}$. Obviously, $S_i \cap T_i=\Phi$ and $S_i \cup T_i=R$. Write S_i as $S_i=\{w_{i,1}, w_{i,2}, \ldots, w_{i,h_j}\}$, where $0 \le w_{i,1} < w_{i,2} < \cdots w_{i,h_i} \le n-1$. Clearly, $\beta_0\beta_i=\sum_{k=1}^{h_i}\beta_{w_{i,k}}$.

Note that for a particular normal basis N, the representation of $\beta_0\beta_i$ is fixed and so is $w_{i,k}$.

Let $\langle x\rangle$ denote the non-negative residue of x mod n. $D=AB$ can be computed by the following identity [1, 3]:

$$\begin{aligned}
D = AB &= \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} a_i b_j \beta_i \beta_j \\
&= \sum_{i=0}^{n-1} a_i b_i \beta_{\langle i+1\rangle} + \sum_{i=1}^{n-1}\sum_{j=0}^{n-1}(a_{\langle i+j\rangle} b_j)(\beta_{\langle i+j\rangle}\beta_j) \\
&= \sum_{i=0}^{n-1} a_i b_i \beta_{\langle i+1\rangle} + \sum_{i=1}^{n-1}\sum_{j=0}^{n-1}(a_{\langle i+j\rangle} b_j)(\beta_i\beta_0)^{2^j} \\
&= \sum_{i=0}^{n-1} a_i b_i \beta_{\langle i+1\rangle} + \sum_{i=1}^{n-1}\sum_{j=0}^{n-1}(a_{\langle i+j\rangle} b_j)\left(\sum_{k=1}^{h_i}\beta_{w_{i,k}}\right)^{2^j} \\
&= \sum_{i=0}^{n-1} a_i b_i \beta_{\langle i+1\rangle} + \sum_{i=1}^{n-1}\sum_{k=1}^{h_i}\left(\sum_{j=0}^{n-1} a_{\langle i+j\rangle} b_j b_{\langle j+w_{i,k}\rangle}\right)
\end{aligned}$$

If we define $B\&A_{n-i}$ as $B\&A_{n-i}=(a_ib_0, a_{\langle i+1\rangle}b_1, \ldots, a_{\langle i+n-1\rangle}b_{n-1})$ and treat it as a field element, then $\sum_{j=0}^{n-1}a_{\langle i+j\rangle}b_j\beta_{\langle j+w_{i,k}\rangle}=(B\&A_{n-i})_{w_{i,k}}$. So we have

$$\begin{aligned}
D &= (B\&A)_1 + \sum_{i=1}^{n-1}\sum_{k=1}^{h_i}(B\&A_{n-i})_{w_{i,k}} \\
&= (B\&A)_1 + \sum_{i=1}^{n-1}\sum_{k\in S_i}(B\&A_{n-i})_k
\end{aligned}$$

Based on this identity and the symmetry of S_i [1, 3], a multiplication algorithm is given in [3].

Let $D=(d_0, d_1, \ldots, d_{n-1})$ be the binary vector of $D=AB$ with respect to the normal basis N, the key function f of N is defined as follows [1]:

$$\begin{aligned}
d_{n-1} &= f(a_0, a_1, \ldots, a_{n-1}; b_0, b_1, \ldots, b_{n-1}) \\
&= a_{n-2}b_{n-2} + \sum_{i=1}^{n-1}\sum_{j=0}^{n-1}\phi_{i,n-1-j}a_{\langle i+j\rangle}b_j
\end{aligned}$$

Recall that S_i is defined as $S_i=\{j|\phi_{i,j}=1\}$. When $k=n-1-j$ runs through S_i, we have

$$f = a_{n-2}b_{n-2} + \sum_{i=1}^{n-1}\sum_{k\in S_i} a_{\langle i+n-1-k\rangle}b_{n-1-k}$$

The circuit complexity of a normal basis multiplier depends on the key function for multiplication. In [2], the complexity of multiplication with respect to the normal basis N is defined as the quantity $C_N=1+\sum_{i=1}^{n-1}h_i$, where $h_i=|S_i|$.

New key function: Recall that the trace function of $A \in GF(2^n)$ over $GF(2)$ is defined as $Tr(A)=\sum_{i=0}^{n-1}A_i$. In particular, $Tr(A)$ equals to the least significant bit of A's Hamming weight in $GF(2^n)$. In software implementations $Tr(A)$ can be found easily in a look-up table. For example, if we create a table with 2^{16} entries on a 32-bit microprocessor, the cost to compute A's Hamming weight is nearly twice that of a field addition operation.

When $B\&A_{n-i}=(a_ib_0, a_{\langle i+1\rangle}b_1, \ldots, a_{\langle i+n-1\rangle}b_{n-1})$ is treated as a field element,

$$\begin{aligned}
Tr(B\&A_{n-i}) = \sum_{k\in R}(B\&A_{n-1})_k &= \sum_{k\in S_i}(B\&A_{n-i})_k \\
&+ \sum_{k\in T_i}(B\&A_{n-i})_k
\end{aligned}$$

Hence, if $|S_i|-|T_i|>c$ (const c depends on the cost to compute $Tr(A)$, for example, $c=2$) then $\sum_{k\in S_i}(B\&A_{n-i})_k$ can be computed faster by the identity:

$$\sum_{k\in S_i}(B\&A_{n-i})_k = Tr(B\&A_{n-i}) + \sum_{k\in T_i}(B\&A_{n-i})_k$$

Now define $\delta_i = \begin{cases} 0 & |T_i| \ge |S_i| \\ 1 & |T_i| < |S_i| \end{cases}$, where $i = 1, 2, \ldots, n-1$.

We have

$$\begin{aligned}
D &= (B\&A)_1 + \sum_{i=1}^{n-1}\sum_{k\in S_i}(B\&A_{n-i})_k \\
&= (B\&A)_1 + \sum_{i=1}^{n-1}\left[(1-\delta_i)\left(\sum_{k\in S_i}(B\&A_{n-i})_k\right)\right. \\
&\qquad \left. + \delta_i\left(Tr(B\&A_{n-i}) + \sum_{k\in T_i}(B\&A_{n-i})_k\right)\right]
\end{aligned}$$

Thus software normal basis multiplication algorithms of [3], which are designed for all normal bases of $GF(2^n)$, can be speeded up by the following method: first, select i's such that $|S_i|-|T_i|>c$ (for example, $c=2$); then for each selected i, compute $\sum_{k\in S_i}(B\&A_{n-i})_k$ using the identity $\sum_{k\in S_i}(B\&A_{n-i})_k=Tr(B\&A_{n-i})+\sum_{k\in T_i}(B\&A_{n-i})_k$. This method saves $|S_i|-|T_i|$ field addition operations for each selected i (excluding computation of $Tr(B\&A_{n-i})$).

In particular, when N is a type I optimal normal basis, the only i satisfying $|S_i|-|T_i|>c$ is $n/2$. In this case, $\beta_0\beta_i=1=\sum_{k=0}^{n-1}\beta_k$, $S_i=\{0, 1, \ldots, n-1\}$ and $|T_i|=0$. Thus n field addition operations (50%) are saved at the cost of a single trace computation (the total number of