DEMITASSE: A “SMALL” VERSION OF THE TINY ENCRYPTION ALGORITHM AND ITS USE IN A CLASSROOM SETTING

JOSHUA HOLDEN

Abstract. This paper describes Demitasse, a cipher which is intended to be used by inexperienced students in a classroom setting. The cipher is based on the Tiny Encryption Algorithm (TEA), but the parameters have been reduced in order to allow the students to operate the cipher with paper and pencil rather than a computer. Nevertheless, students using the cipher should gain some understanding into the inner workings of a modern computer cipher. An example of the use of Demitasse in the classroom is also discussed.

1. Introduction

In May 2010, while I was on sabbatical, I taught a general education mathematics course at Goshen College in northern Indiana, using the topic of cryptography. The course was entitled “Mathematical World” and the idea was to introduce students to practical applications of mathematics without requiring more than a very basic amount of high school algebra as a prerequisite. I did the usual additive and affine ciphers, exponential ciphers and RSA, and basic cryptanalysis, among other topics, but I also wanted to show the students something resembling a modern block cipher of the sort that might actually be implemented on a modern digital computer. In addition, I wanted them to be able to do a hands-on paper and pencil exercise with a cipher of this sort. In past courses I had talked about DES, the Digital Encryption Standard, using S-DES, the simplified version of DES invented by Ed Shaefer and presented in [17]. However, since it is now fairly well established that DES is insecure, I felt that something invented more recently was in order.

The first time I taught a course on cryptography for nontechnical students (see [6] for details) I used the sections on linear and nonlinear feedback shift registers from [1, Sections 3.4–3.5] as my examples of modern digital ciphers, but I really felt like the students were missing something by not giving them a proper block cipher. In the cryptography course I teach at my home institution (see [6] for more on this course also) I have for the last several years talked about AES, the Advanced Encryption Standard, using S-AES, the simplified version of AES from [14]. That is a class for math and science majors, however. I felt that while technically AES can be grasped by anyone who understands modular arithmetic and the manipulation of polynomials, it would be pushing it awfully hard for a class with the very low prerequisite level of the Goshen class.

2. TEA

While I was thinking about other possibilities for a cipher to use, I remembered that one year during my class at Rose-Hulman, a student had given a presentation on TEA, the Tiny Encryption Algorithm [23]. As [21, p. 49] points out, there is a trade-off in the design of an iterated block cipher. One can have a complicated round and do fewer rounds, like AES, or one can have a simple (but still nonlinear) round, and do it very many times, or somewhere in the middle. TEA was invented in 1994 and was designed to have a simple round repeated very many times. This simple round structure makes it easy to constructed a scaled-down version of TEA which can be done by hand by a student. Another design goal of TEA was ease of software implementation [12], especially on 32-bit architectures [21, p. 49]. This led to the choice of using arithmetic modulo 2^{32} as one of its basic operations and eschewing the use of S-boxes and complicated permutations in favor of arithmetic and logical operations and bit shifts [12]. TEA also has an extremely simple key schedule, which was another attractive feature for using it as the basis of a pencil and paper exercise. For more information about TEA, see, in addition to the above references, the survey paper [18] and the TEA Web Archive [19].

TEA uses a key size of 128 bits and a block size of 64 bits. It is very nearly a Feistel cipher, although addition modulo 2^{32} is used to combine the round function with the block rather than addition modulo 2. This means that the decryption function is slightly different from the encryption function, although both are so simple that the difference is not generally problematic. The Feistel rounds are grouped into pairs, called cycles. A full encryption consists of 64 rounds, or 32 cycles. Two rounds (one cycle) of the encryption algorithm are illustrated in Figure 1, where $\boxplus$ represents addition modulo 2^{32} and $\oplus$ represents bitwise addition modulo 2. Unlike DES and many other Feistel ciphers, in the last round the last swap is done, just like in the other rounds. This again makes decryption slightly different from encryption, but since a separate implementation for decryption is necessary anyway

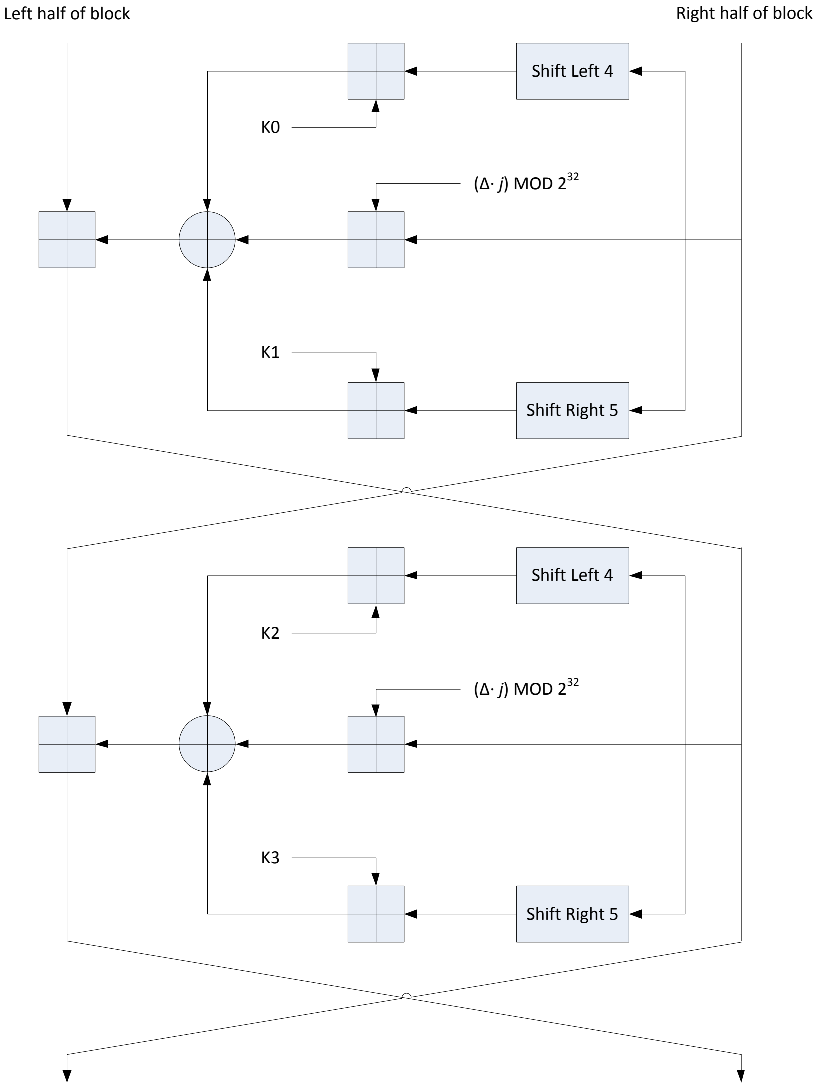


FIGURE 1. Two rounds (one cycle) of TEA encryption

the difference is worthwhile to simplify the description. Decryption is the same except that the two additions modulo 2^{32} at the left-hand side are replaced with subtractions modulo 2^{32}, and the swaps are done at the start of the round instead of the end.

As mentioned earlier, the designers of TEA chose to rely on addition modulo 2^{32}, bitwise addition modulo 2, and bit shifts for its basic operations rather than the more common S-boxes and permutations. This is reasonable because addition modulo 2^{32} and bitwise addition modulo 2 are in some sense “incompatible” operations. According to [18], Don Coppersmith and Jim Massey independently showed that mixing such incompatible operations achieves the same goals as using S-boxes and permutations. In particular, [5] and [20] studied the particular combination of addition modulo 2^n and bitwise addition modulo 2 and [25] studied combinations of these operations with bit rotations. The paper [25] shows that sufficiently complicated combinations of these operations can in some cases produce a large fraction of all possible mappings from plaintext to ciphertext and therefore can be reasonable replacements for S-boxes and permutations. The best-known cipher using this technique is IDEA, proposed in 1991 by Lai and Massey. (The first version of IDEA appears in [11] and the final version in [10].) It uses the operations of addition modulo 2^{16}, bitwise addition modulo 2, and multiplication modulo $2^{16}+1$. In [11], it is shown that these operations are incompatible in a certain specified sense. TEA replaces the modular multiplication with bit shifts and compensates for the reduced complexity with more rounds, according to the philosophy described at the beginning of this section.

The key schedule for TEA is extremely simple. The 128-bit key is divided into four 32-bit words, denoted K0, K1, K2, and K3 in Figure 1. Each cycle uses the four subkeys in an identical manner. In order to make the rounds distinct, a round constant is also used. This consists of the “magic number”

$$\Delta = \mathrm{9E3779B9}_{16} = \lfloor (\sqrt{5}-1)2^{31} \rfloor$$

multiplied by the number of the current round (denoted by j on the diagram) modulo 2^{32}. Of course, the subkeys and round constants have to be applied in the appropriate order during decryption.

In fact, the designers eventually decided the key schedule was too simple and in 1996 invented the eXtended Tiny Encryption Algorithm, or XTEA [15] to overcome a related-key attack and the fact that each key is equivalent to three others. The same paper which presented XTEA also introduced Block TEA, which is a new mode of operation for XTEA and similar ciphers. A flaw in Block TEA was subsequently discovered, and Corrected Block TEA, also known as XXTEA [16] was released in 1998. An attack against XXTEA was announced in 2010 [24]. I chose not to consider XTEA and XXTEA in the development of my simplified cipher.

An article about TEA would not be complete without a mention of the most famous (ab)use of the cipher. The so-called "version 1.1" of the Microsoft Xbox gaming console, introduced in 2002, used a hash function based on the TEA cipher to prevent the use of unauthorized software. Unfortunately (from Microsoft's point of view), the existence of equivalent keys allowed Xbox users to make a small but crucial change in the Microsoft boot code which completely bypassed the security [3, 22].[1]

3. Demitasse

So after a bit of a refresher on TEA, I decided to come up with a simplified version of the algorithm for my class, along the lines of S-DES and S-AES. The simplified algorithm uses a reduced key size (16 bits instead of 128), a reduced block size (8 bits instead of 64) and a reduced number of rounds (4 rounds instead of 64). Originally I did call it S-TEA (Simplified TEA), but I recently learned that Fauzan Mirza also has an algorithm called Simplified TEA [12], so I renamed it to avoid confusion. (Mirza's algorithm uses 64-bit keys and the same values as TEA for the block size and number of rounds, but it has a much simplified and hence less secure round function and an even simpler key schedule than TEA. It was designed to teach specifically about cryptanalysis rather than about ciphers in general. You should think of Mirza's algorithm as "Weak TEA" whereas mine is a "small cup" of TEA.)

The structure of Demitasse is exactly the same as TEA. The differences are in the key size (16 bits instead of 128), the block size (8 bits instead of 64) and the number of rounds (4 rounds instead of 64). (The number of rounds was chosen as the minimum that would illustrate the iterative nature of the cipher, and does not really capture well the trade-off between simplicity and repetition — perhaps this should be emphasized when teaching using the cipher.) The rounds are grouped into cycles just as in TEA. Operations are done on 4-bit words rather than 32-bit words, so the additions and subtractions are modulo 16. The "magic number" is

$$\Delta = 9_{16} = 1001_2.$$

Other than that, everything is just like in TEA.

[1] Incidentally, the moral of this story is not so much to avoid ciphers with equivalent keys. The moral is really "Do your homework." TEA was never designed to be used in a hash function, and in fact experts had warned in 1996 that a hash function based on TEA would be insecure [8].

The course at Goshen college was a three-week "May term" course which met four days a week with most days consisting of a one-hour lecture, a one-hour break, and then a two-hour block which was usually divided into about an hour of lecture and an hour of hands-on work. By the day that we discussed Demitasse, the students had quite a bit of experience with hands-on exercises including shift and multiplicative ciphers, exponential ciphers, and RSA. The first hour of the day was an introduction to binary numbers and their operations, starting with how to convert back and forth between binary and decimal numbers. I then discussed binary addition, subtraction, multiplication, and (long) division, before moving on to "addition without carry" (bitwise addition modulo 2), "addition ignoring the overflow" (addition modulo 2^n) and "subtraction ignoring the overborrow" (subtraction modulo 2^n). I pointed out that "subtraction without borrow" was the same as "addition without carry", and finally I illustrated left and right shifts. Somewhat to my surprise, the students seemed to have a fairly easy time with this material. Many of the students either had seen the material before or caught on quite quickly (I'm not sure which), and they appeared to have a good time helping the students who were taking longer to catch on.

After the break I talked briefly about DES, AES, TEA, and Feistel ciphers, and then gave the students the handout that appears (with corrections) in Appendix A. After a very short explanation, I started them working in pairs on the exercises from the handout. Most students made it through the encryption exercise by the end of the two-hour block, with various amounts of help from me and a lot of cross-checking between different groups (which I encouraged). A few students made it through the decryption exercise as well. (Those that did discovered that I had made an error in the order of the subkeys on the decryption schedule! That has been fixed on the version included here.)

4. Security and Cryptanalysis of Demitasse

As I mentioned, the security of each round of TEA, and therefore of Demitasse, relies on the mixing of incompatible operations. Because of the very low level of mathematical and computational sophistication of the students in my class, I did not discuss this idea in any detail. Likewise, I have not yet discussed cryptanalysis of Demitasse in a classroom setting. The inventors of S-DES and of S-AES, however, used their creations to discuss cryptanalysis as well as cipher design. It would certainly be worthwhile to do this for Demitasse as well.

In terms of known attacks, first of all, Demitasse has the same issue with equivalent keys that was pointed out for TEA in [8]. Since adding $8 = 1000_2$ modulo 16 is the same as adding 1000_2 modulo 2 bitwise, adding this number to both K_0 and K_1, or both K_2 and K_3, or all four of the above, will cancel out. Thus each Demitasse key has three other equivalent keys, and the effective key space has size 2^{14} rather than 2^{16}.

Mirza discusses a number of possible attacks on TEA and his simplified version of it, starting with some very trivial ones. He notes first that an iterated cipher with only two rounds succumbs to a trivial attack based on the inherent structure of a Feistel cipher [12, Section 5], and an iterated cipher of three or fewer rounds is vulnerable to a meet-in-the-middle attack [12, Section 6]. At four rounds, Demitasse is at least resistant to these attacks.

I do not know of any published linear cryptanalysis of TEA. Mirza sketches a linear cryptanalysis of a reduced-round version of his simplified version of TEA in [12, Section 8]. He notes that because the linear approximation needs to follow the carry from the addition modulo 2^{32}, linear cryptanalysis will be ineffective beyond 31 cycles, or 62 rounds. In our case this suggests that linear cryptanalysis of Demitasse might be possible up to 3 cycles, or 6 rounds. Whether this is in fact the case is not clear. (At least, not to me.)

Differential cryptanalysis of the full TEA cipher appears to be difficult. Mirza [12, Section 10] sketches a differential cryptanalysis of an 8-round version of his simplified cipher using a 6-round differential characteristic. Hong, et al. [7], however, suggest that the combination and order of operations used in TEA make it difficult to apply a differential attack. They do give a truncated differential attack against a 17-round version of TEA based on an 8-round truncated differential characteristic. Moon, et al. [13] had earlier given an impossible differential cryptanalysis of 11-round TEA based on a 10-round impossible differential characteristic. It would be interesting to see how these attacks play out against Demitasse. I suspect that they would be very feasible against the four rounds of standard Demitasse and perhaps against 8 or even 16 rounds.

Several more attacks are possible exploiting the very simple key schedule of TEA. Fleming [4] showed that a version of TEA which uses Δ instead of $\Delta \cdot j$ in every round was vulnerable to an attack based on the birthday paradox. (See [9, Footnote 4] for comments on this attack.) This general type of attack is now known as a "slide attack" [2], and should also be applicable to a similarly weakened Demitasse. Kelsey, et al. [9] give three related-key attacks against TEA. The first two seem like they would be very easy to adapt to Demitasse and the third one (which is more complicated) might work also.

5. Conclusion

My goal for this project was to show my students a cipher that they could do with pencil and paper, which would still be representative of a modern computer cipher. At the very least it should operate on strings of binary digits, use operations common to computer ciphers, and have a structure resembling at least one computer cipher used in practice. I believe that Demitasse fulfilled those goals. First of all, students did successfully work with a bit string cipher within a (two-hour) class period! Without that, this project would not have been a success.

Demitasse uses operations that appear not only in TEA but in lots of other ciphers. Bitwise addition modulo 2 (or exclusive-or), of course, probably appears in almost every computer cipher, including DES and AES. Addition modulo 2^n does not appear in either DES or AES, but it does appear in other well-known ciphers, such as IDEA (used in several versions of PGP), MARS, RC6, and Twofish (three of the AES finalist ciphers). (Subtraction appears in most of these as well.) Bit shifts appear in at least some implementations of AES, and play a larger role in ciphers such as Serpent (another AES finalist). (Several other ciphers mentioned above, including DES, use bit rotations rather than bit shifts.) On a more abstract level, the idea of mixing incompatible operations is central not only in TEA but also in IDEA, as mentioned in Section 2, and it provides part (but not all) of the security in MARS, RC6, and Twofish.

In terms of larger structure, Demitasse, like TEA, has for all practical purposes the structure of a Feistel cipher. This is an extremely common technique in modern cipher design, with DES being one of the earliest and certainly the best-known example. In addition, Twofish has almost precisely a Feistel structure and MARS and RC6 have generalizations of the structure.

Considering all of this, I was extremely pleased with the success of Demitasse! I hope that you will also find it useful in your classrooms.

Acknowledgments

My thanks to the anonymous reviewer for helpful suggestions and references.

References

1. Beutelspacher, A. 1994. *Cryptology*, Washington, DC: Mathematical Association of America.
2. Biryukov, A. and Wagner, D. 1999. *Slide Attacks.* Fast Software Encryption: 6th International Workshop (Rome, Italy, March 24–26, 1999), Proceedings (Knudsen, L., ed.), LNCS, vol. 1636, Springer-Verlag, Berlin, Heidelberg, pp. 245–259.
3. Broersma, M. October 14, 2002. *New Xbox security cracked by Linux fans.* ZDNet UK / News and Analysis / Application Development. Available at `http://www.zdnet.co.uk/news/application-development/2002/10/14/new-xbox-security-cracked-by-linux-fans-2123851/`.
4. Fleming, R. October 22, 1996. *An attack on a weakened version of TEA.* sci.crypt Usenet group. Available at `https://groups.google.com/group/sci.crypt/browse_frm/thread/4a19a18cbdcf78eb`. Web page includes a reply by David Wagner from October 24, 1996 with an alternative version of the attack.
5. Grossman, E. February 26, 1974. *Group Theoretic Remarks on Cryptographic Systems Based on Two Types of Addition*, Technical Report RC 4742, IBM Research Report, Yorktown Heights, N.Y.
6. Holden, J. 2004. *A Comparison of Cryptography Courses.* Cryptologia. 28(2): 97–111, DOI 10.1080/0161-110491892809.
7. Hong, S., Hong, D., Ko, Y., Chang, D., Lee, W., and Lee, S. 2004. *Differential Cryptanalysis of TEA and XTEA*, Information Security and Cryptology — ICISC 2003 (Lim, J. and Lee, D., eds.), LNCS, vol. 2971, Springer-Verlag, Berlin, Heidelberg, pp. 402–417.
8. Kelsey, J., Schneier, B., and Wagner, D. 1996. *Key-Schedule Cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES*, Advances in Cryptology – CRYPTO '96 (Koblitz, N., ed.), LNCS, vol. 1109, Springer-Verlag, Berlin, Heidelberg, pp. 237–251.
9. ———. 1997. *Related-key cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA.* Information and Communications Security: First International Conference, ICIS '97 (Beijing, China, November 11–14, 1997), Proceedings (Han, Y., Okamoto, T., and Qing, S., eds.), LNCS, vol. 1334, Springer-Verlag, Berlin, Heidelberg, pp. 233–246.
10. Lai, X., *On the design and security of block ciphers*, Doctoral Dissertation, Swiss Federal Institute of Technology, Zurich, Switzerland, 1992, `http://e-collection.library.ethz.ch/eserv/eth:38650/eth-38650-02.pdf`.
11. Lai, X. and Massey, J. L. 1991. *A Proposal for a New Block Encryption Standard*, Advances in Cryptology – EUROCRYPT '90 (Damgård, I. B., ed.), LNCS, vol. 473, Springer-Verlag, Berlin, Heidelberg, pp. 389–404.
12. Mirza, F. March 1998. *Block Ciphers And Cryptanalysis*, available at `http://fmirza.seecs.nust.edu.pk/papers/report.pdf`.
13. Moon, D., Hwang, K., Lee, W., Lee, S., and Lim, J. 2002. *Impossible Differential Cryptanalysis of Reduced Round XTEA and TEA.* Fast Software Encryption: 9th International Workshop (Leuven, Belgium, February 4–6, 2002), Proceedings (Daemen, J. and Rijmen, V., eds.), LNCS, vol. 2365, Springer-Verlag, Berlin, Heidelberg, pp. 49–60.

14. Musa, M. A., Schaefer, E. F., and Wedig, S. 2003. *A Simplified AES Algorithm and its Linear and Differential Cryptanalyses.* Cryptologia. 27(2): 148–177, DOI 10.1080/0161-110391891838.
15. Needham, R. M. and Wheeler, D. J. October 1997. *Tea Extensions*, Notes, available at `http://www.cix.co.uk/~klockstone/xtea.pdf`.
16. ______. October 1998. *Correction to xtea*, Notes, available at `http://www.cix.co.uk/~klockstone/xxtea.pdf`.
17. Schaefer, E. F. 1996. *A Simplified Data Encryption Standard Algorithm.* Cryptologia. 20(1): 77–84, DOI 10.1080/0161-119691884799.
18. Shepherd, S. J. 2007. *The Tiny Encryption Algorithm.* Cryptologia. 31(3): 233–245, DOI 10.1080/01611190601090606.
19. ______. *The TEA Web Archive.* `http://www.simonshepherd.com/tea.htm`. Accessed Sep. 12, 2011.
20. Staffelbach, O. and Meier, W. 1991. *Cryptographic Significance of the Carry for Ciphers Based on Integer Addition*, Advances in Cryptology – CRYPTO '90 (Menezes, A. J. and Vanstone, S. A., eds.), LNCS, vol. 537, Springer-Verlag, Berlin, Heidelberg, pp. 602–614.
21. Stamp, M. 2005. *Information Security: Principles and Practice*, Hoboken, NJ: Wiley-Interscience.
22. Steil, M. 2005. *17 Mistakes Microsoft Made in the Xbox Security System.* 22nd Chaos Communication Congress: Private Investigations (Berlin, December 27). Electronic proceedings, available at `http://events.ccc.de/congress/2005/fahrplan/attachments/591-paper_xbox.pdf`.
23. Wheeler, D. J. and Needham, R. M. 1995. *TEA, a Tiny Encryption Algorithm.* Fast Software Encryption: Second International Workshop (Leuven, Belgium, December 14–16, 1994), Proceedings (Preneel, B., ed.), LNCS, vol. 1008, Springer-Verlag, Berlin, pp. 363–366.
24. Yarrkov, E. 2010. *Cryptanalysis of XXTEA*, Technical Report 254, Cryptology ePrint Archive, International Association for Cryptologic Research (IACR), available at `http://eprint.iacr.org/2010/254`.
25. Zieschang, T. 1997. *Combinatorial Properties of Basic Encryption Operations*, Advances in Cryptology – EUROCRYPT '97 (Fumy, W., ed.), LNCS, vol. 1233, Springer-Verlag, Berlin, Heidelberg, pp. 14–26.

Appendix A. Class Handout

Demitasse: A Simplified TEA Algorithm

Joshua Holden, Rose-Hulman Institute of Technology

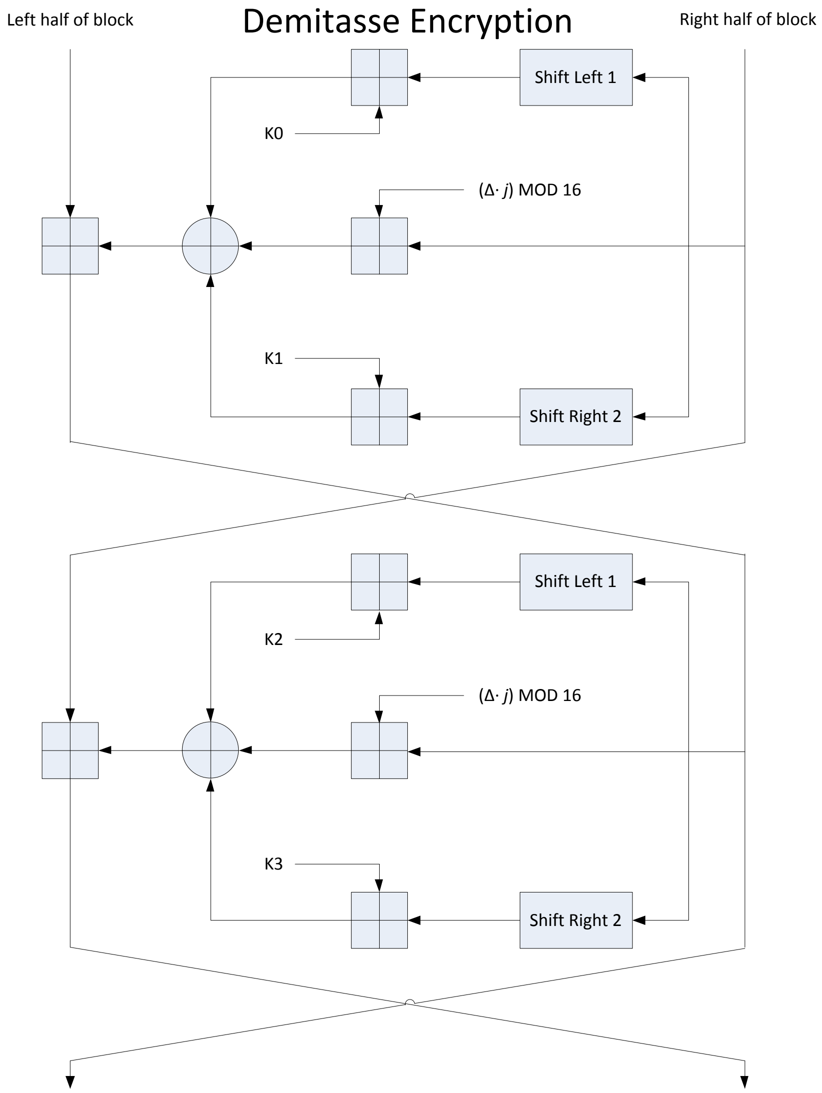
Overview. Demitasse is to TEA as S-DES is to DES. In fact, the structure of Demitasse is exactly the same as TEA. The differences are in the key size (16 bits instead of 128), the block size (8 bits instead of 64) and the number of rounds (4 rounds instead of 64). The rounds are grouped in 2 pairs, called cycles.

Encryption. The next page shows a picture of two rounds (one cycle) of Demitasse encryption. Demitasse encryption uses four types of operations on binary numbers: addition with carry (ignoring the overflow), addition without carry, left shift, and right shift. In each round the block is divided into two 4-bit halves. The right half is acted on by some operations and then added with carry to the left half. The halves are then swapped and the process repeats.

On the diagram:
$\boxplus$ is addition with carry (ignoring the overflow)
$\oplus$ is addition without carry

The operations on the right half involve each of the four operations, the key, and a magic constant. The key is divided into four 4-bit sections, K_0, K_1, K_2, and K_3. Each cycle uses K_0 and K_1 in the first half and K_2 and K_3 in the second half. There is also a magic constant, Δ, which in Demitasse equals 9 (decimal) or 1001 (binary). In round j ($j = 1$ through 4), the constant $(\Delta \cdot j) \operatorname{MOD} 16$ is used.

Exercise. Use the key 1011 0101 1010 0110 to encrypt the plaintext “ah” as expressed in binary, that is 0110 1000.

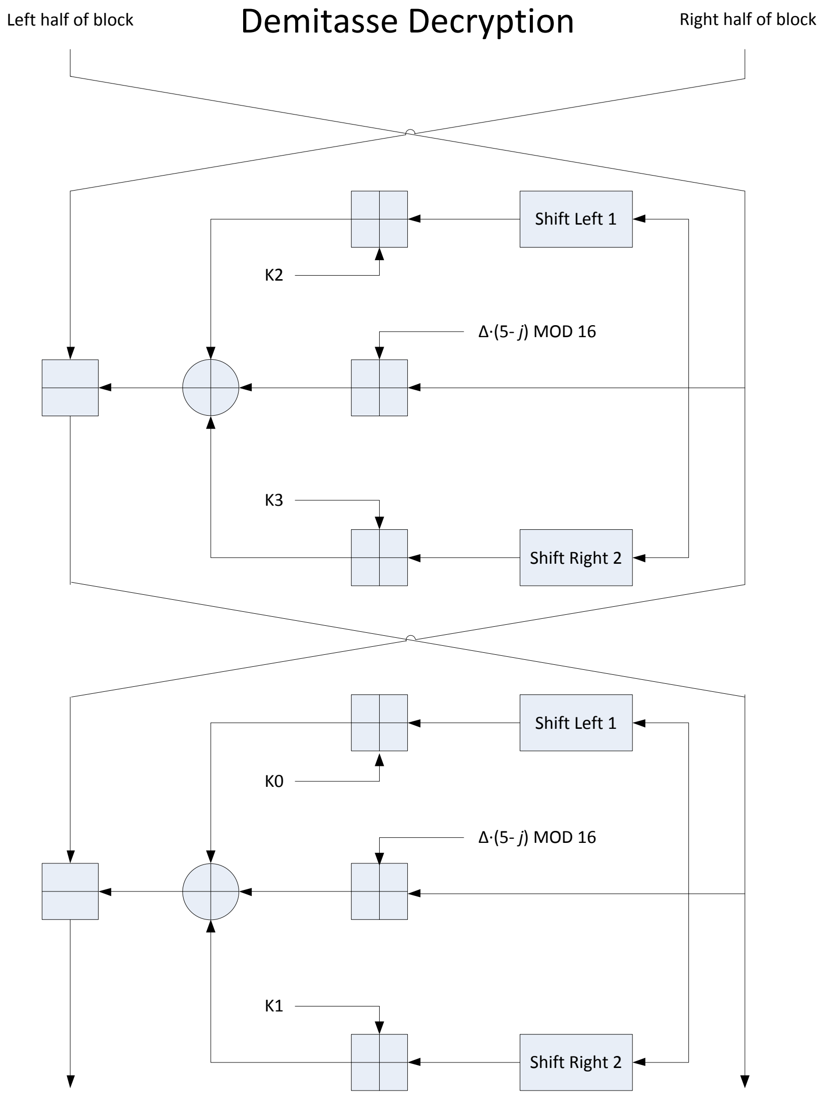

Demitasse Encryption
Left half of block
Right half of block
Shift Left 1
K0
(Δ· *j*) MOD 16
K1
Shift Right 2
Shift Left 1
K2
(Δ· *j*) MOD 16
K3
Shift Right 2

Decryption. Demitasse decryption is almost the same, except:

- **One** place in each round, subtraction with borrow ignoring the “overborrow” is used instead of addition with carry ignoring the overflow,
- the key sections and constants are used in a different order, and
- the swaps are done at the beginning of the round instead of the end.

  On the diagram:
  $\boxplus$ is addition with carry (ignoring the overflow)
  $\boxminus$ is subtraction with borrow (ignoring the “overborrow”)
  $\oplus$ is addition without carry

Exercise. Decrypt your answer from the previous exercise. Did you get your plaintext back?


Left half of block
Demitasse Decryption
Right half of block
Shift Left 1
K2
Δ·(5- j) MOD 16
K3
Shift Right 2
Shift Left 1
K0
Δ·(5- j) MOD 16
K1
Shift Right 2

Biographical Sketch

Joshua Holden is currently an Associate Professor in the Mathematics Department of Rose-Hulman Institute of Technology, an undergraduate engineering college in Indiana. He received his Ph.D. from Brown University in 1998 and held postdoctoral positions at the University of Massachusetts at Amherst and Duke University. His research interests are in computational and algebraic number theory, cryptography, and the application of graph theory to fiber arts. His teaching interests include the use of technology in teaching and the teaching of mathematics to computer science majors, as well as the use of historically informed pedagogy. His non-mathematical interests used to include fiber arts, but that now seems to be a mathematical interest. Still largely in the non-mathematical category are his interests in science fiction and music, both classical and contemporary. He currently plays drums and sings backup in the band Whisper Down.

Department of Mathematics, Rose-Hulman Institute of Technology, Terre Haute, IN 47803, USA
E-mail address: holden@rose-hulman.edu