

# Fast Digital Convolutions using Bit-Shifts

Shekhar S. Chandra

**Abstract**—An exact, one-to-one transform is presented that not only allows digital circular convolutions, but is free from multiplications and quantisation errors for transform lengths of arbitrary powers of two. The transform is analogous to the Discrete Fourier Transform, with the canonical harmonics replaced by a set of cyclic integers computed using only bit-shifts and additions modulo a prime number. The prime number may be selected to occupy contemporary word sizes or to be very large for cryptographic or data hiding applications. The transform is an extension of the Rader Transforms via Carmichael's Theorem. These properties allow for exact convolutions that are impervious to numerical overflow and to utilise Fast Fourier Transform algorithms.

**Index Terms**—DSP-FAST; Number Theoretic Transform; Discrete Fourier Transform; Fast Fourier Transform; Fermat Number Transform.

## I. INTRODUCTION

The Discrete Fourier Transform (DFT) is commonly used to compute the circular convolution  $h$  of two finite (or periodic) sequences  $f$  and  $g$  of length  $N$  as

$$h(j) = f(j) * g(j) = \sum_{k=0}^{N-1} f(k) \cdot g(j-k), \quad (1)$$

by using the Convolution Theorem, where Eq. (1) can be computed simply as a product of both sequences in Discrete Fourier space. This theorem provides a computational advantage because the Cooley-Tukey algorithm [1] for computing the DFT has a computational complexity of  $O(N \log_2 N)$ , as opposed to  $O(N^2)$  for direct methods, when  $N$  is a power of two.

A major result of this letter regarding convolutions can be summarised as follows. Let  $\langle a \rangle_m$  denote computing the remainder with respect to  $m$  (see Appendix A for details), where  $a \in \mathbb{Z}$ , i.e.  $a$  is an integer, and  $m$  is a prime number (or prime) as given in, but not restricted to, Table I. To compute the digital circular convolution of two finite integer sequences, one transforms both sequences as

$$X(u) = \sum_{t=0}^{N-1} \langle x(t) \cdot 2^{ut} \rangle_m, \quad (2)$$

which only involves bit-shifting, modulo and addition operations. The coefficients of these two sequences are multiplied and the result is inverted as

$$x(t) = \frac{1}{N} \sum_{u=0}^{N-1} \langle X(u) \cdot 2^{-ut} \rangle_m, \quad (3)$$

where  $2^{-ut} = 2^{(-ut)_N} = 2^{N-ut}$ . Note that the convolution is free from round-off errors as no floating-point numbers are

required. Exact digital filtering involving division operations can be performed via multiplicative inverses, i.e. an integer  $N^{-1}$  so that  $\langle N \cdot N^{-1} \rangle_m = 1$ . The Cooley-Tukey algorithm [1] is easily applied by replacing  $e^{2\pi i \alpha / N}$  with the powers of two  $2^\alpha$ , where  $\alpha \in \mathbb{Z}$  and  $i^2 = -1$ . In other words, the  $N^{\text{th}}$  root of unity  $e^{2\pi i N} = 1$  is replaced with the integer-only equivalent  $\langle 2^N \rangle_m = 1$ . The transform lengths permitted are  $N = 2^n$  when  $n \in \mathbb{Z}$  that divide  $N_{\text{Max}}$  in Table I. For example, the prime 13631489 allows for all  $N$  up to and including  $2^{19}$ .

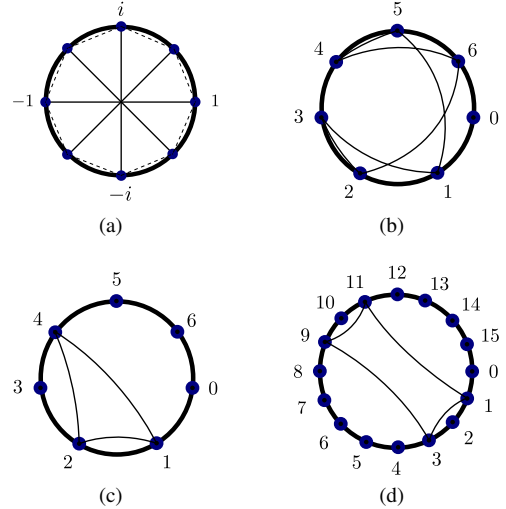


Figure 1. The unit circle (a) and the digital circles for computing exact convolutions (b)-(d). The paths in (b)-(d) show the integers produced from successive powers of 3 modulo 7 (resulting in  $\{1, 3, 2, 6, 4, 5, 1, \dots\}$ ) in (b), successive powers of 2 modulo 7 (resulting in  $\{1, 2, 4, 1, 2, 4, 1, \dots\}$ ) in (c) and successive powers of 3 modulo 16 (resulting in  $\{1, 3, 9, 11, 1, \dots\}$ ) in (d). The transform lengths permissible for each are  $N_{\text{Max}} = 6$  (and its divisors) for (b),  $N_{\text{Max}} = 3$  for (c) and  $N_{\text{Max}} = 4$  for (d).

$m$ Prime	Max. Transform Length $N_{\text{Max}}$	Corresponding Fermat Number	Word Size
641	64	$F_5$	16-bit
2424833	1024	$F_9$	32-bit
319489	4096	$F_{11}$	32-bit
13631489	524288	$F_{18}$	32-bit

Table I  
PRIME VALUES OF THE MODULUS  $m$  FOR DIFFERENT TRANSFORM LENGTHS  $N$ . LENGTHS PERMISSIBLE ARE  $2^n \leq N_{\text{MAX}}$ . NOTE THAT THE SIZE OF THE PRIME IS NOT NECESSARILY PROPORTIONAL TO  $N_{\text{MAX}}$ .

Equations (2) and (3) are an extension of the Rader Transforms (RTs) [2], which, until now, were only practical for small ( $N \leq 128$ ) transform lengths, severely limiting its applications [3]. The theory developed in Sec. II and III of this letter applies Carmichael's Theorem to remove these limitations

completely by generalising the concept of an integer based  $N^{\text{th}}$  root of unity. Table I shows that the moduli  $m$  chosen for very large transform lengths  $N$  can fit into a 32-bit word size because the primes are shown to be any (including the smallest) one of the factors of large Fermat numbers, which are numbers of the form

$$F_n = 2^{2^n} + 1. \quad (4)$$

Sec. III also presents a modulus free transform similar to Eqs (2) and (3), i.e. an integer-only transform without the need for modulo operations, using the same theory.

The preservation of the Circular Convolution Property (CCP), which allows one to use the Convolution Theorem for finite sequences, is made possible because the unit circle of the DFT (see Fig. 1(a)) is replaced with the digital “circle”

$$2^N \equiv 1 \pmod{m}, \quad (5)$$

so that  $2^N - 1$  is a multiple of  $m$  (see Figs 1(c) and 2). The successive powers  $\{1, \dots, N\}$  of two generates a unique set of integers in some order modulo  $m$ . The result is a circle consisting of a set of cyclic integers, i.e. a set of unique integers with a period. These integers define the “harmonics” of the transform in Eqs (2) and (3). Integer coefficients allow computations to be done without round-off error or numerical overflow, since the results are congruent modulo  $m$  [4].

The new transform is an extension of the RTs [2], which consist of the Fermat Number Transform (FNT) [5] and the Mersenne Number Transform (MNT) [6]. When using bit-shifts, the FNT and the MNT only utilise moduli 64-bits or less in size for small ( $N \leq 128$ ) transform lengths. The FNT only allows larger transform lengths when not in bit-shift form, i.e. the sequence is multiplied by powers of integers other than two (such as three) in Eqs (2) and (3). A solution is to use multi-dimensional techniques which provide a limited extension of the transform lengths [3]. Pollard [7] showed that these transforms may have an alternate form (that also does not utilise bit-shifting) via Euler’s Theorem

$$a^N \equiv a^{\phi(m)} \equiv 1 \pmod{m}, \quad (6)$$

where  $a, m \in \mathbb{Z}$ ,  $a$  and  $m$  are coprime and  $\phi(m)$  is provided in Appendix B. When  $m$  is prime, Eq. (6) becomes  $a^{m-1} \equiv 1 \pmod{m}$ , which is known as Fermat’s (little) Theorem. These alternate transforms also preserve the CCP while allowing arbitrary transform lengths and are referred to as Number Theoretic Transforms (NTTs). The modulus  $m$  for these transforms are primes of the form  $p = k \cdot N + 1$  so that  $m - 1 = k \cdot N$ , allowing transform lengths of  $N = 2^n$  and divisors of  $N$  [8]. On the rare occasions when computations explicitly require large exponents in Eqs (5) and (6), they can still be computed in logarithmic complexity using modular exponentiation methods.

The NTTs are ideal for real data but can also be complex-valued [9]. NTTs have been applied to fast multiplication of very large integers [5], fast digital convolutions and filtering [2, 3], encryption [10] and discrete Radon transforms [11]. Agarwal and Burrus [2] showed the NTT to be faster than the Fast Fourier Transform (FFT) in their implementation.

Chandra [11] (via the open-source library [12]) showed that a modern implementation of the NTT outperforms the popular FFTW library.

## II. CARMICHAEL’S THEOREM

This section presents a new and more general theory of Number Theoretic Transforms (NTTs) utilising the concept of primitive roots from Carmichael’s Theorem [13] (see Appendix B), a generalisation of Euler’s Theorem given in Eq. (6). The primary result of this new theory are Eqs (2) and (3) when using Table I.

To construct an NTT, one needs a set of unique cyclic integers sufficient to represent all the coefficients of a given transform length  $N$ . In Euler’s Theorem, the integer  $a$  is a special integer called a primitive root (or  $\phi$ -root [13]) related to the modulus, where successive powers  $\{1, \dots, m-1\}$  of  $a$  generates all the integers  $\{1, \dots, m-1\}$  in some unique order modulo  $m$  (see Fig. 1(b)). This condition works well, but is very restrictive as not all integers are  $\phi$ -roots of a given modulus and not all moduli have  $\phi$ -roots. For example, the integer 2 is only a  $\phi$ -root for primes of the form  $4 \cdot q + 1$  when  $q$  itself is prime [14, pg. 102]. Thus, the integer 2 is only suitable for prime length NTTs and not a  $\phi$ -root of primes of the form  $k \cdot 2^n + 1$  required for power of two transform lengths in this theory.

Carmichael [13] developed the concept of the primitive  $\lambda$ -root, where the successive powers  $\{1, \dots, \nu\}$  of this root generates a fixed subset of the integers  $\{1, \dots, m-1\}$  in some unique order modulo  $m$  (see Fig. 1(c)). The number of integers in this subset is  $\nu$ , where  $\nu$  is the smallest integer for which

$$a^\nu \equiv 1 \pmod{m}, \quad (7)$$

is true. Thus, one gets a set of unique cyclic integers of order  $\nu$  capable of representing  $\nu$  distinct coefficients. Carmichael [13] points out the smallest composite (non-prime) modulus  $m$  for when this and Eq. (5) is true is  $m = 341$ . Since  $341 = 11 \cdot 31$  and  $2^{10} - 1 = 3 \cdot 11 \cdot 31$ , then  $\nu = 10$  as  $2^{10} \equiv 2^{340} \equiv 1 \pmod{341}$ . Such composite moduli are now known as Poulet numbers. To construct a unique and sufficient set of coefficients for power of two transform lengths, one needs to show that  $\nu = 2^n$  and find a modulus  $m$  so that 2 is a  $\lambda$ -root of  $m$ .

## III. NEW TRANSFORMS

This section presents the proof of the transform stated in Eqs (2) and (3). It will be shown that the primes in Table I for this transform can be chosen to be any (including the smallest) prime factor of the Fermat numbers (4). The section will conclude with a discussion of another useful result.

### A. Multiplication Free

In order to satisfy Eq. (5), a prime modulus must be selected so that

$$m \mid (2^{2^n} - 1), \quad (8)$$

i.e.  $m$  is a prime factor of  $2^{2^n} - 1$ , when the transform length  $N$  is a power of two. The modulus is chosen to be prime so that one may divide the coefficients by any integer, allowing

the construction of arbitrary exact filters. The value  $2^{2^n} - 1$  must be the smallest multiple of  $m$  to the base 2 so that 2 is a  $\lambda$ -root of  $m$  as given by (7) and Sec. II.

The numbers of the form  $2^{2^n} - 1$ , which will be denoted as the Rader numbers, are a specific form of the Mersenne numbers

$$M_n = 2^n - 1. \quad (9)$$

Mersenne numbers can always be expressed as

$$2^{ab} - 1 = (2^a - 1) \left( 1 + 2^a + 2^{2a} + \dots + 2^{(b-1)a} \right), \quad (10)$$

when  $n$  is composite, since they are binomial numbers [15, pg. 42]. Applying this expansion to the Rader numbers

$$\begin{aligned} n = 1 : & \quad 2^2 - 1 \\ n = 2 : & \quad 2^4 - 1 = (2^2 - 1) \cdot (1 + 2^2) \\ n = 3 : & \quad 2^8 - 1 = (2^2 - 1) \cdot (1 + 2^2 + 2^4 + 2^8) \\ & \quad = (2^2 - 1) \cdot (1 + 2^2) \cdot (1 + 2^4) \\ n = 4 : & \quad 2^{16} - 1 = (2^2 - 1) \cdot (1 + 2^2 + 2^4 + 2^6 + \dots + 2^{14}) \\ & \quad = (2^2 - 1) \cdot (1 + 2^2 + 2^4 + 2^8) \cdot (1 + 2^8) \\ & \quad = (2^2 - 1) \cdot (1 + 2^2) \cdot (1 + 2^4) \cdot (1 + 2^8), \end{aligned}$$

and so on until one arrives at an identity of the Fermat numbers [16, pg. 26]

$$2^{2^n} - 1 = F_0 \cdot F_1 \cdot F_2 \cdot \dots \cdot F_{n-1} = F_n - 2. \quad (11)$$

For example, the first several factorisations of  $2^n - 1$  are

$$\begin{aligned} 2^2 - 1 &= 3, & 2^3 - 1 &= 7, \\ 2^4 - 1 &= 3 \cdot 5, & 2^5 - 1 &= 31, \\ 2^6 - 1 &= 3^2 \cdot 7, & 2^7 - 1 &= 127, \\ 2^8 - 1 &= 3 \cdot 5 \cdot 17, & 2^9 - 1 &= 7 \cdot 73, \\ 2^{10} - 1 &= 3 \cdot 11 \cdot 31, & 2^{11} - 1 &= 23 \cdot 89. \end{aligned}$$

Eq. (11) suggests that  $m$  should be a Fermat number, noting that only the first five Fermat numbers are known to be prime.

**Proposition 1** (Fermat Number Moduli). *The smallest power of two  $\nu$  in Eq. (7) for which*

$$2^{2^n} \equiv 1 \pmod{m}, \quad (12)$$

*is true, is when the modulus  $m$  is the Fermat number  $F_{n-1}$ .*

*Proof:* By the identity (11), higher order Fermat numbers can only become a factor of a given Rader number as  $n$  increases. Multiples of these higher order Fermat numbers cannot exist as factors of Mersenne numbers  $2^\ell - 1$  less than the given Rader number since the only divisor of  $2^n$  is two and powers of two by Eq. (10), i.e.  $\ell$  cannot be any number other than the divisors of  $2^n$ . Hence by Eq. (11),  $2^{2^n} - 1$  is the smallest multiple of  $F_{n-1}$  to the base 2 so  $N = 2^n$ . ■

This results in a reformulation of the FNT, which is only suitable for small transform lengths as the Fermat numbers grow large rapidly and are composite after  $F_4$ . Can one extend the above theorem to include the prime factors of large Fermat numbers? The answer is yes and it is the main theoretical result of this letter.

**Proposition 2** (Fermat Factor Moduli). *The smallest power of two  $\nu$  for which Eq. (12) is true, is when the modulus  $m$  is a factor  $r$  (prime or otherwise) of the Fermat number  $F_{n-1}$ .*

*Proof:* Assume the contrary, that there exists a Mersenne number  $2^\ell - 1$  less than a given Rader number that is also a multiple of  $r$ . From Eq. (10),  $\ell \mid 2^n$ , but the only divisors of  $2^n$  is two or its powers. Now it is well known that the Fermat numbers do not share any common factor with each other, i.e. they are pair-wise coprime [15, pg. 63]. Hence,  $\ell$  cannot be a power of two less than  $2^n$  and so the first multiple of  $2^\ell - 1$  must be the Rader number. Thus, for  $F_5 = 641 \cdot 6700417$ , 641 or 6700417 allows for  $N = 2^n$  when  $n \leq 6$ , since neither prime divides  $2^\ell - 1$  for all  $\ell < 2^6$ . ■

We denote  $r$  as a Rader prime when  $r$  is prime. Some useful Rader primes are given in Table I, resulting in the transforms given in Eqs (2) and (3). This is a far more useful result than Prop. 1 as now the moduli may be small or as large as desired, by simply selecting a Fermat number with a suitable small or large Rader prime. Although the factorisation of Fermat numbers is still an active area of research [17], there are sufficient numbers of factors already known to accommodate any word size or for data hiding via large moduli.

Applying the Generalised Fermat and Mersenne Numbers to Prop. 2 should extend the NTT of Dimitrov *et al.* [18]. Euler [19] showed that all prime factors of the Fermat numbers are of the form  $k \cdot 2^{n+2} + 1$ , so Prop. 2 should also extend the work of Bhattacharya and Astola [8]. Finally, hardware implementations of this new transform may be similar to those constructed by McClellan [20] and Leibowitz [21] for the FNT, since both require simple bit-shifting and prime factors of the Fermat numbers are of the form  $k \cdot 2^{n+2} + 1$  [19]. See Agarwal and Burrus [2, Sec.VI.E] for examples of how to compute the bit-shifting. The next section introduces an integer-only transform not requiring modulus operations.

## B. Modulus Free

Carmichael [13] also proves the useful result

$$a^{2^{\alpha-2}} \equiv 1 \pmod{2^\alpha}, \quad (13)$$

where  $a$  and 2 are coprime (see Appendix B). This result can be used in constructing transforms that preserve the CCP, while requiring no modulo operations in programming languages (such as *C*) or architectures that support “wrap around” upon overflow, i.e. the act of truncation is equivalent to modulo power of two (see Fig. 1(d)). This is advantageous because the integer division instruction, a critical part of the modulo operation, is generally a slow instruction. Note that an expression like Eq. (13) is not possible using Euler’s Theorem [13].

Normalisation of the transform is also a concern, since the multiplicative inverse does not exist. This can be resolved by ensuring  $2^\alpha$  is sufficiently large so that unnormalised values do not exceed  $2^\alpha$ . In other words, if  $2^\beta$  is the bit depth of the data, then  $\alpha \geq \beta + N$ .

The implementations of these transforms can be found in the NTTW C library [12]. Applications and performance comparisons of the various NTTs, as well as to the DFT, will be part of a future publication.

## CONCLUSION

Transforms for fast digital convolutions were constructed that did not either require any multiplications or modulo operations (see Eqs (2), (3), Table I and (13)). The former utilises only bit-shifts, additions and modulo operations on prime factors of the Fermat numbers (denoted as Rader primes), while the latter only uses multiplications and additions. The result was made possible by using Carmichael's generalisation of Euler's Theorem, which also provides a more general theory of Number Theoretic Transforms.

## ACKNOWLEDGEMENTS

My thanks go to the Faculty of Science and School of Physics, Monash University and to Dr. I. Svalbe, Dr B. Parrein, Dr N. Normand and G. Ruben for their comments.

## APPENDIX A

## CONGRUENCES AND SINO NOTATION

Sino notation for computing the remainder or modulo operation is given as  $\langle a \rangle_m$ , which denotes  $a \pmod{m}$ , i.e.  $a = \langle a \rangle_m + m \cdot q$ , with  $a, \langle a \rangle_m, q, m \in \mathbb{Z}$  so that  $0 \leq \langle a \rangle_m < m$ . The modulo operation allows one to define a congruence, an example of which is given in Fig. 2. Multiplicative inverses  $b^{-1}$ , i.e. the equivalent integers within congruences to do division by  $b$  (thus turning division into a multiplication), are found using the efficient Extended Euclidean algorithm, provided they are coprime or their greatest common divisor is one, i.e.  $\text{gcd}(b, m) = 1$ .

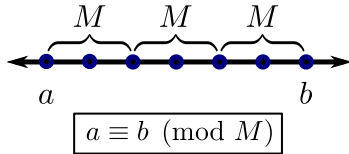


Figure 2. Integer  $a$  is congruent to integer  $b$  when the distance between them is a multiple of  $M$ .

## APPENDIX B

## TOTIENT &amp; LAMBDA FUNCTIONS

The Totient function  $\phi(m)$  is the number of integers less than  $m$  that do not have a common factor (or are coprime) with  $m$ . For example, when  $m$  is prime, the function  $\phi(m) = m - 1$ . The Lambda function  $\lambda(m)$  is defined in terms of the Totient function as follows

$$\lambda(m) = \begin{cases} \phi(p^\alpha), & \text{if } m = p^\alpha \text{ and } p \text{ is an odd prime} \\ \phi(2^\alpha), & \text{if } m = 2^\alpha \text{ and } 0 \leq \alpha < 2 \\ \frac{1}{2}\phi(2^\alpha), & \text{if } m = 2^\alpha \text{ and } \alpha > 2 \end{cases},$$

so that  $\lambda(2^\alpha p_1^{\alpha_1} \dots p_j^{\alpha_j})$  is the lowest common multiple of  $\lambda(2^\alpha), \lambda(p_1^{\alpha_1}), \dots, \lambda(p_j^{\alpha_j})$  for  $m = 2^\alpha p_1^{\alpha_1} \dots p_j^{\alpha_j}$ . This allows one to define Carmichael's Theorem [13]

$$a^{\lambda(m)} \equiv 1 \pmod{m}. \quad (14)$$

## REFERENCES

- [1] J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, 1965.
- [2] R. Agarwal and C. Burrus, "Fast convolution using Fermat number transforms with applications to digital filtering," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 22, no. 2, pp. 87–97, Apr 1974.
- [3] —, "Fast one-dimensional digital convolution by multidimensional techniques," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 22, no. 1, pp. 1–10, Feb 1974.
- [4] H. J. Nussbaumer, "Overflow detection in the computation of convolutions by some number theoretic transforms," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 26, no. 1, pp. 108–109, Feb 1978.
- [5] A. Schönhage and V. Strassen, "Fast multiplication of large numbers," *Computing*, vol. 7, no. 3-4, pp. 281–292, 1971, in German.
- [6] C. M. Rader, "Discrete Convolutions via Mersenne Transforms," *Computers, IEEE Transactions on*, vol. C-21, no. 12, pp. 1269–1273, Dec. 1972.
- [7] J. M. Pollard, "The Fast Fourier Transform in a Finite Field," *Mathematics of Computation*, vol. 25, no. 114, pp. 365–374, 1971.
- [8] M. Bhattacharya and J. Astola, "Number theoretic transform modulo  $k \cdot 2^N + 1$ , a prime," vol. 4, Tampere, Finland, 2000, pp. 2529 – 32.
- [9] H. J. Nussbaumer, "Complex convolutions via Fermat number transforms," *IBM Journal of Research and Development*, vol. 20, no. 3, pp. 282–284, 1976.
- [10] S. Mohan and B. Adiga, "Fast algorithms for implementing RSA public key cryptosystem," *Electronics Letters*, vol. 21, no. 17, pp. 761–, 15 1985.
- [11] S. S. Chandra, "Exact image representation via a Number-Theoretic Radon Transform," *IET Computer Vision*, vol. In Peer Review, submitted April, 2010.
- [12] —, "The Finite Transform Library (FTL) featuring the NTTW library," Monash University, Australia, C Library (Open Source GPL v3), 2009. [Online]. Available: <http://finitetransform.sourceforge.net>
- [13] R. D. Carmichael, "Note on a new number theory function," *Bulletin of the American Mathematical Society*, vol. 16, no. 5, pp. 232–238, 1910.
- [14] A. Beiler, *Recreations in the Theory of Numbers - The Queen of Mathematics Entertains*. Dover Publications Inc., 1966.
- [15] P. Schumer, *Mathematical Journeys*. Wiley-Interscience, 2004.
- [16] M. Křížek, F. Luca, and L. Somer, *17 lectures on Fermat numbers : from number theory to geometry*. Springer, 2001.
- [17] R. Brent, "Factorization of the tenth and eleventh Fermat numbers," Australian National University, Tech. Rep., 1996.
- [18] V. Dimitrov, T. Cooklev, and B. Donevsky, "Generalized Fermat-Mersenne number theoretic transform," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 41, no. 2, pp. 133–139, Feb 1994.
- [19] L. Euler, "Observationes de theoremate quodam Fermatiano aliisque ad numeros primos spectantibus," *Acad. Sci. Petropol.*, vol. 6, pp. 103–107, 1732.
- [20] J. McClellan, "Hardware realization of a Fermat number transform," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 24, no. 3, pp. 216–225, Jun 1976.
- [21] L. Leibowitz, "A simplified binary arithmetic for the Fermat number transform," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 24, no. 5, pp. 356–359, Oct 1976.