

論文2003-40SD-2-4

# 양자화롭과 오차롭을 사용한 직접 디지털 주파수 합성기

## (A Direct Digital Frequency Synthesizer Using Quantization ROM And Error ROM)

梁炳燾\*, 成基赫\*, 金榮俊\*, 金利變\*

(Byung-Do Yang, Ki-Hyuk Sung, Young-Joon Kim, and Lee-Sup Kim)

## 요 약

새로운 직접 디지털 주파수 합성기(DDFS)가 제안되었다. 제안된 DDFS는 기존의 DDFS에서의 각 롬(ROM)들을 양자화롭과 오차롭으로 나누어 저장하는 새로운 롬 압축 방식을 사용한다. 제안된 DDFS에서의 전체 롬 크기는 기존의 롬에 비하여 상당히 줄어들었다. 12비트 출력 데이터를 가지는 DDFS의 경우, 롬 압축률은 78분의 1에 이른다. 성능 검증을 위하여 사인 함수의 12비트 출력 데이터를 가지는 DDFS가 0.35um CMOS 공정으로 구현되었다. 3.3V 전원과 100MHz 클럭에서의 소모 전력은 9.56mW이고 최고 동작 클럭 주파수는 330MHz이다.

## Abstract

A new direct digital frequency synthesizer (DDFS) is proposed. The DDFS uses a new ROM compression method that divides each ROM in the conventional DDFS into two ROMs (a quantization ROM and an error ROM). The total size of the ROMs in the proposed DDFS is significantly reduced compared to the original ROM. The ROM compression ratio of 78 is achieved for a DDFS with 12bit output data. A DDFS with 12bit output data for sine function was implemented in a 0.35um CMOS technology. The power dissipation is 9.56mW at 100MHz with 3.3V and the maximum operating clock frequency is 330MHz.

**Keywords** : frequency synthesizer, ROM

## I. 서 론

주파수 합성기(frequency synthesizer)는 디지털 통신에서 널리 사용되고 있는 핵심 부품으로 기존 주파수

\* 正會員, 韓國科學技術院 電子電算學科

(Dept. of Electrical Engineering & Computer Science, KAIST)

※ 본 연구는 한국과학재단과 미세정보센터의 지원을 받아 수행되었습니다. 칩 제작에 도움을 주신 한국 전자통신연구원에 감사 드립니다.

接受日字:2002年2月15日, 수정완료일:2003年1月30日

를 이용하여 원하는 새로운 주파수를 만드는 회로이다. 주파수 합성기를 구현하기 위하여 phase locked loop (PLL)을 기반의 주파수 합성기들이 주로 사용되었다. PLL 기반의 주파수 합성기들은 높은 주파수 선택성을 가지는 사인파를 만들어 낸다. 그러나, 확장 스펙트럼 랜(spread spectrum LAN), 군사용 주파수 변환 시스템(military frequency hopping system) 그리고 PCS와 같은 디지털 통신 시스템들은 빠른 주파수 변환, 정밀한 주파수 해상도, 넓은 주파수 대역 그리고 높은 주파수 선택성을 필요로 한다. PLL 기반의 주파수 합성기들은 본질적으로 피드백 루프(feedback loop)를 가지고 있어서 빠른 주파수 변환을 할 수 없다. 따라서, 빠른 주파수 변환과 높은 주파수 해상도를 요구하는 통신

시스템에는 PLL 기반의 주파수 합성기들은 적합하지 않다. 이런 요구들을 만족시키기 위하여 새롭게 등장한 개념이 직접 디지털 주파수 합성기 (direct digital frequency synthesizer: DDFS) 이다.

기존의 DDFS는 롬(read only memory: ROM)을 사용하여 사인함수를 저장한다. 더 높은 주파수 선택성을 위해서는 더 많은 데이터가 롬에 저장되어야 한다. 또한, 더 넓은 대역의 주파수를 만들어 내기 위해서는 더 높은 클럭(clock) 주파수에서 동작해야 한다. 그러나, 더 많은 데이터를 저장하기 위하여 사용되는 큰 롬은 DDFS의 크기와 소모 전력을 증가시키고 최고 동작 클럭 주파수를 낮춘다. 따라서, DDFS의 성능을 높이고 크기와 소모 전력을 낮추기 위해서는 롬의 크기를 줄이는 것이 매우 중요하다. 이를 위하여, 롬의 크기를 줄이는 많은 기술들이 제안되었다<sup>1-7)</sup>.

이 논문에서는 새로운 롬 압축 방식을 사용한 새로운 저전력 DDFS가 제안되었다. 제안된 롬 압축 방식은 기존의 DDFS에서 사용되는 각 롬들을 양자화롭과 오차롭으로 나누어 저장함으로써 그 크기를 줄이는 방식이다.

II. 기본 구조

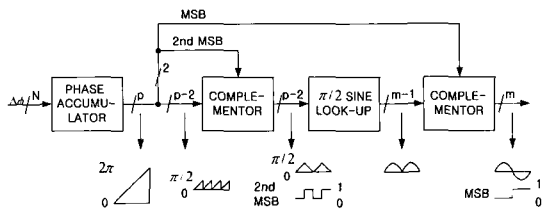


그림 1. DDFS 기본 구조  
Fig. 1. Basic structure of the DDFS.

<그림 1>은 DDFS 기본 구조의 간단한 블록 다이어그램을 보여준다. 매 클럭마다 N비트 입력 주파수 Δφ가 N비트 위상 누적기에 더해진다. 위상 누적기에서의 p비트 위상 출력은 위상값에 따른 m비트 사인 값을 저장하고 있는 롬의 어드레스로 사용된다. DDFS의 주파수 해상도는 Δf = f<sub>clk</sub>/2<sup>N</sup> 이고 DDFS의 출력 주파수는 f<sub>clk</sub> = Δf × Δφ이다. n번째 클럭에서 p비트 위상 누적기의 출력이 φ(n)일 때, DDFS의 출력은 sin(2πφ(n)/2<sup>N</sup>) 이 된다.

<그림 1>의 DDFS는 사인함수의 대칭성을 사용하여 2πrad 대신 π/2rad의 사인 값만을 저장하는 방식을 사용하였다. 최상위비트는 DDFS 출력 값의 사인비트를 결정하고 두 번째 상위비트는 사인파의 증감을 결정한다. 위상 누적기 출력의 나머지 p-2비트는 사인 롬의 어드레스로 사용된다.

사인함수를 저장하기 위한 롬의 크기는 매우 크다. 그 롬의 크기는 (m-1) × 2<sup>p-2</sup>가 된다. 32비트 입력 주파수와 12비트 출력 값을 가지는 DDFS의 경우, 45K비트 롬이 필요하다.

롬의 크기를 줄이는 한 방식은 삼각함수의 특성을 이용한 Sunderland 방식이다<sup>11)</sup>. 이 방식은 출력에서 1/2 LSB(least significant bit) 오프셋(offset)을 발생시키지만, 롬 크기를 상당히 줄일 수 있는 방식이다. 사인파의 위상 어드레스가 φ(n) = α + β + γ로 나누어진다고 가정하면, 12비트 위상 값은 α < 1, β < 2<sup>-4</sup>, γ < 2<sup>-8</sup>의 3개의 4비트 값들로 분할된다. 이때, 사인 함수는 다음과 같이 표현된다.

$$\begin{aligned} & \sin\left(\frac{\pi}{2}(\alpha + \beta + \gamma)\right) \\ &= \sin\left(\frac{\pi}{2}(\alpha + \beta)\right)\cos\left(\frac{\pi}{2}\gamma\right) + \cos\left(\frac{\pi}{2}(\alpha + \beta)\right)\sin\left(\frac{\pi}{2}\gamma\right) \\ &\approx \sin\left(\frac{\pi}{2}(\alpha + \beta)\right) + \cos\left(\frac{\pi}{2}\alpha\right)\sin\left(\frac{\pi}{2}\gamma\right) \end{aligned}$$

이 방식은 12비트 어드레스를 가지는 하나의 롬을 8비트 어드레스를 가지는 두개의 작은 롬들(coarse 롬과 fine 롬)로 나눔으로써 전체 롬 크기를 줄일 수 있는 방식이다.

사인위상차 알고리즘은 coarse 롬에 저장된 값을 줄이기 위하여 제안되었다. 사인 값 대신 사인 값과 위상의 차이를 롬에 저장함으로써, 롬의 저장될 값을 크기를 줄여 롬의 크기를 줄일 수 있다<sup>12)</sup>. 그러나 이 방식은 롬에 저장되어 있는 사인위상차와 위상을 더하여 사인 값을 만들기 위한 덧셈기가 필요하다.

III. 새로운 롬 압축 방식

새롭게 제안된 롬 압축방식은 Sunderland 알고리즘과 사인위상차 알고리즘에 의하여 만들어진 두 개의 롬들의 크기를 줄여준다. 두 개의 롬들(coarse 롬과 fine 롬)을 압축하기 위해서, 각 롬은 다시 한번 두 개

의 롬들로 나누어 진다. 이렇게 나누어진 두 롬들이 양자화롭(quantization ROM)과 오차롭(error ROM)이다. <그림 2>는 새로운 알고리즘을 적용한 DDFS의 블록 다이어그램이다. 양자화롭은 오리지널 롬에 저장된 데이터를 양자화시킨 상위 몇 비트만을 저장하고, 오차롭은 오리지널 롬 데이터와 양자화롭 데이터의 차이를 저장한다.

$2^k$  어드레스 롬 데이터가  $2^i$  섹션으로 균등하게 나누어졌을 때, 양자화롭은 각 섹션마다 최소값 하나만을 저장한다. 각 섹션에서 저장되는 값은 q비트로 양자화된 값이다. 따라서, 양자화롭의 어드레스는  $2^i$ 에서  $2^k$ 로 줄어들고, 양자화롭의 크기는  $2^i \times q$ 가 된다. 오차롭은 오리지널 롬과 양자화롭 사이의 오차 값을 저장한다. 이 오차 값들은 오리지널 롬에 저장되어 있던 데이터의 크기보다 매우 적다. 오차 값들 중 가장 큰 값을 저장하기 위하여 e비트가 필요하다고 가정하면, 오차롭의 크기는  $2^k \times e$ 비트가 된다.

<그림 3(a)>에서와 같이, coarse롬과 fine롬에 저장된 데이터들은 롬 어드레스가 증가함에 따라 부드럽게 변한다. 같은 섹션의 데이터들은 그 값이 비슷하여 각 섹션 안에서 데이터들의 최대 차이는 매우 적다. 따라서, 오차롭의 크기는 오리지널 롬의 크기에 비하여 매우 작게 된다.

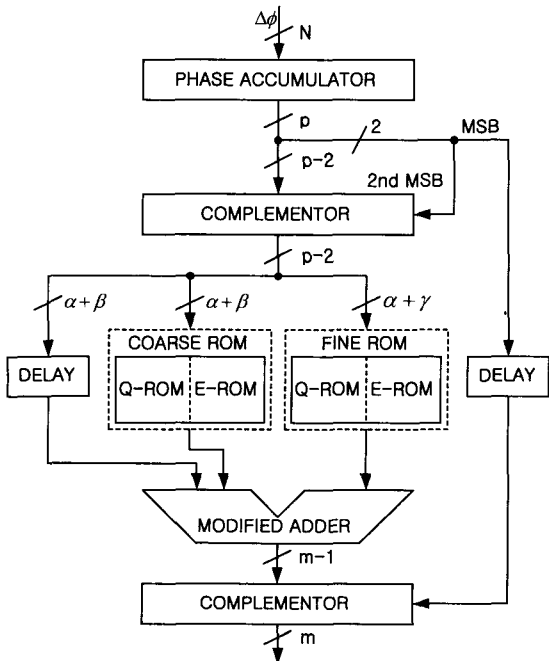
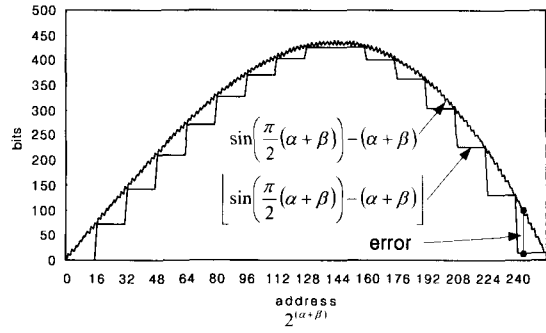
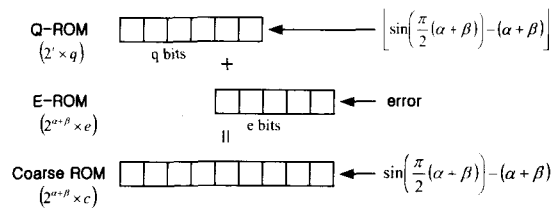


그림 2. 제안된 DDFS의 블록 다이어그램  
Fig. 2. Block diagram of the proposed DDFS.



(a)



(b)

그림 3. coarse 롬의 양자화롭과 오차롭 데이터  
(12비트 출력 DDFS에서 16섹션, (=4, (=4)  
Fig. 3. Q-ROM and E-ROM data in coarse ROM.  
(16 sections, (=4, (=4 in the 12bit output DDFS)

우 작게 된다. 또한, 양자화롭은 섹션 수가 오리지널 롬의 어드레스에 비하여 작고 저장되는 데이터도 상위 몇 비트만을 저장하므로, 양자화롭도 오리지널 롬의 크기에 비하여 매우 작게 된다. 결과적으로 양자화롭과 오차롭을 합한 전체 롬 크기도 오리지널 롬보다 작게 된다.

<그림 3>은 12비트 출력을 가지는 DDFS의 coarse 롬 데이터를 저장하고 있는 양자화롭과 오차롭을 보여 준다. 양자화롭은 각 섹션의 양자화된 최소값을 저장하는데, 롬의 크기를 줄이기 위하여 coarse롬의 출력 비트가 아닌 q비트만을 저장한다. 양자화롭과 오차롭의 데이터를 저장하는 전체 크기 ( $2^i \times q + 2^k \times e$ bits)를 최소화하기 위하여, 섹션 수( $2^i$ ), 양자화롭에 저장되는 비트( $q$ ), 그리고 모든 오차 데이터를 저장하기 위한 최소 비트( $e$ )의 최적의 값을 찾아야 한다. 파라미터들( $i, q, e$ )은 다음과 <그림 4>의 알고리즘을 사용하여 찾을 수 있다.

(i) 수가  $2^i$ 이고 오리지널 롬의 입력 어드레스가  $2^k$  일 때,  $i = k$ .

- (ii)  $q$  = 오리지널 롬의 출력 비트
- (iii) 양자화롬에 저장될 각 섹션의  $q$ 비트 최소값들을 찾는다.
- (iv) 오리지널 롬과 양자화롬의 오차값들을 찾는다.
- (v) 모든 오차를 표현하기 위한 최소 비트  $e$ 를 찾는다.
- (vi) 전체 롬 크기( $2^i \times q + 2^k \times e$ )를 계산한다.
- (vii)  $q = 1$ 이면 (viii)로 이동하고, 아니면 (iii) - (vi)를 반복한다.
- (viii)  $i = 1$ 이면 (ix)로 이동하고, 아니면 (ii) - (vii)를 반복한다.
- (ix) (vi)의 결과를 사용하여 전체 롬 크기가 최소인 최적의  $i, q, e$ 를 찾는다.

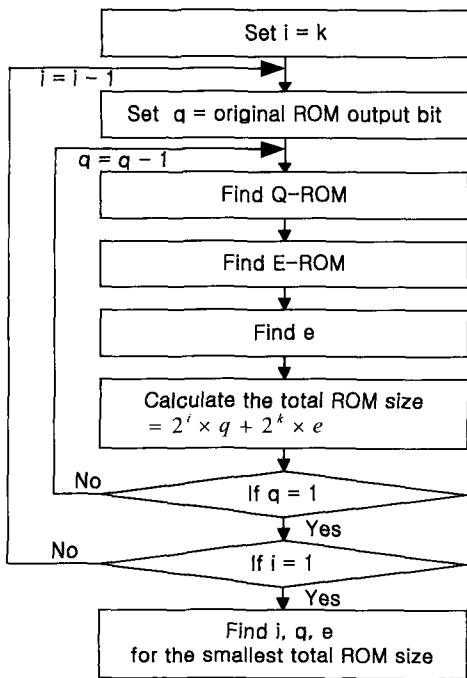


그림 4. 롬 파라미터들을 찾는 알고리즘  
Fig. 4. Algorithm of finding ROM parameters.

표 1. 제안된 DDFS의 롬 크기  
Table 1. ROM size of the proposed DDFS.

		12bit output	10bit output
Coarse ROM	Q-ROM	$2^2 \times 6$	$2^2 \times 4$
	E-ROM	$2^3 \times 5$	$2^2 \times 4$
Fine ROM	Q-ROM	$2^2 \times 2$	$2^2 \times 1$
	E-ROM	$2^3 \times 3$	$2^2 \times 2$
Total ROM size		2496	928

표 2. 롬 크기 비교

Table 2. ROM size comparison.

Compression Technique	ROM size in normalized bits	
	12bit output	10bit output
Uncompressed memory	78.77	44.1
Sunderland architecture [1]	1.54	1.65
Nicholas architecture [3]	1.23	1.38
Proposed architecture	1	1

<표 1>은 DDFS의 출력이 12비트와 10비트 일 때의 제안된 DDFS에서의 롬 크기를 보여준다. <표 2>는 기존 DDFS에서의 롬들의 상대적인 크기들은 보여준다. 각 롬의 크기는 제안된 DDFS의 롬에 크기로 나누어졌다. 결과적으로, 제안된 DDFS는 12비트와 10비트의 출력을 가지는 경우에 Nicholas 알고리즘<sup>[3]</sup>에 비하여 각각 19%와 28%의 롬 크기의 감소를 이루었다.

IV. 제안된 DDFS의 구조

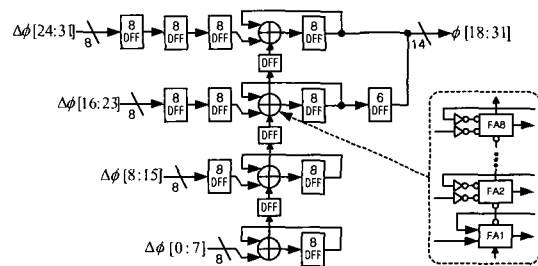


그림 5. 위상 누적기  
Fig. 5. Phase accumulator.

<그림 5>은 8비트 리플 캐리 덧셈기(ripple carry adder)를 사용하여 4 단 파이프라인 구조로 만든 32비트 위상 누적기이다. 8비트 리플 캐리 덧셈기는 8개의 전가산기(full adder)를 사용하여 구조를 단순화하였다.

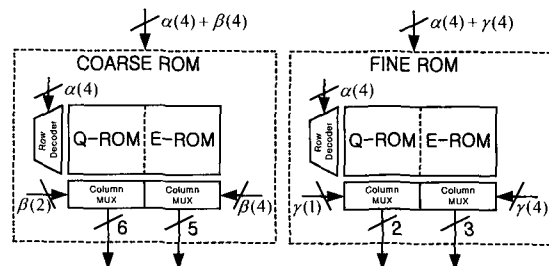
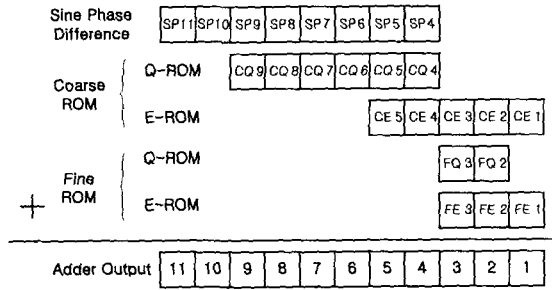
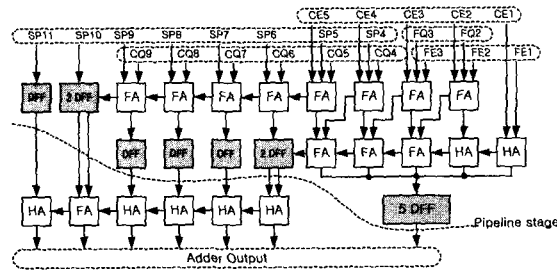


그림 6. 양자화롬과 오차롬  
Fig. 6. Q-ROM and E-ROM.



(a)



(b)

그림 7. 수정된 덧셈기  
Fig. 7. Modified adder.

<그림 6>의 양자화롭과 오차롭의 구조를 보면, 양자화롭과 오차롭은 실제 레이아웃에서 두 개의 롬으로 나누어 지지 않는다. 양자화롭과 오차롭은 일정 부분 같은 어드레스 비트를 공유하기 때문에 같은 row decoder로 두 롬의 데이터를 얻을 수 있다. 따라서, 하나의 롬 안에 양자화롭과 오차롭의 데이터를 저장하여 구현할 수 있다. 그러나 양자화롭과 오차롭의 어드레스가 완전하게 일치하는 것은 아니므로 두 개의 다른 column MUX를 사용해야 한다. 따라서, 하나의 롬을 양자화롭과 오차롭으로 분할 저장하는 것이 실제 하드웨어 구현에서 롬 수를 증가시키지 않는다.

<그림 7>은 4개의 롬 출력 데이터와 위상값을 더해 DDFFS의 출력 데이터를 만들기 위해서 수정된 덧셈기이다. 제안된 DDFFS에서는 속도를 높이고 크기를 줄이기 위하여 두 단의 파이프라인을 가진 더해지는 데이터들에 적합하게 수정된 덧셈기를 사용하였다. 기존의 덧셈기를 사용할 경우, 5개의 다른 데이터를 더하기 위해서 4개의 덧셈기가 필요하고 고속의 동작 속도를 얻기 위해서 많은 수의 파이프라인 단이 필요하다. 그러나, 각 파이프라인 단을 증가시키기 위해서는 DFF(Data flip-flop)을 추가적으로 넣어야 한다. 이는 칩 면적과 소모 전력을 증가시킨다. 수정된 덧셈기는 롬의 출력들과 사인위상차의 입력을 재배치함으로써, 덧셈기의 수

를 줄였다. 또한, 덧셈기 내부에서의 데이터의 흐름과 파이프라인을 적절하게 조절함으로써, 덧셈기에서 발생하는 캐리 리플이 5개의 전가산기(full adder) 이내로 발생하도록 수정하였다. 수정된 덧셈기는 덧셈기 입력을 재배치하고 전가산기와 반가산기(half adder)를 적절히 사용함으로써 크기를 줄이고 속도를 향상시켰다.

V. 칩 제작 및 실험 결과

제안된 DDFFS구조는 0.35um 공정을 사용하여 제작되었다. 32비트 입력 주파수와 12비트 출력을 가지는 제작된 DDFFS의 성능이 <표 3>에 요약되어 있다. 칩 테스트 결과, 최고 동작 클럭 주파수는 330MHz였다. 3.3V의 전원 전압과 100MHz의 클럭 주파수에서 수행한 HSPICE 시뮬레이션에서의 소모전력은 9.56mW였다. DDFFS의 출력의 FFT 테스트는 -86.9 dBc 이상의 주파수 특성을 가짐을 보여 주었다. 제작된 DDFFS 칩의 트랜지스터 수는 7,168개이고 면적은 550 × 220 um<sup>2</sup>이다.

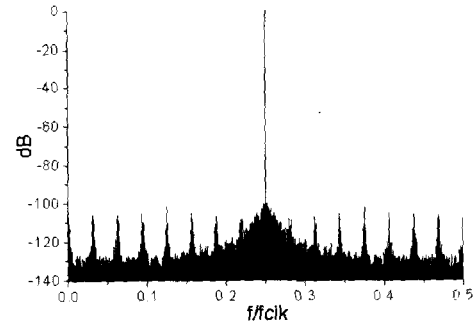


그림 8. FFT를 수행한 출력 주파수 스펙트럼  
Fig. 8. Output frequency spectrum by FFT.

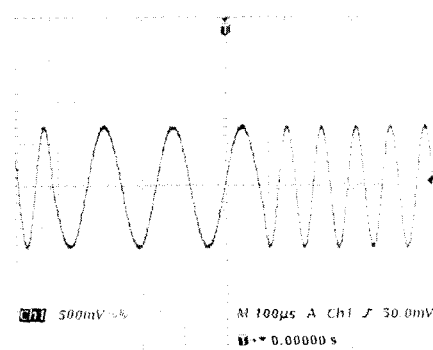


그림 9. D/A converter의 출력 파형  
Fig. 9. Output waveforms from D/A converter.

제작된 DDFS칩의 동작 검증은 logic analyzer로 DDFS 칩의 출력을 얻어 수행되었다. <그림 8>은 logic analyzer로 얻은 DDFS 칩의 디지털 출력을 FFT를 수행한 디지털 출력 주파수 스펙트럼이고, <그림 9>는 DDFS 칩의 출력을 D/A converter를 통하여 얻은 아날로그 파형이다. <그림 10>은 제작된 DDFS 칩 사진을 보여준다.

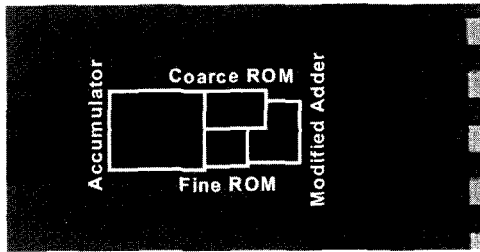


그림 10. DDFS 칩 사진  
Fig. 10. Photograph of the DDFS chip.

표 3. DDFS 칩 사양  
Table 3. Specification of the DDFS chip.

Frequency Input Word	32 bits
Output Word Length	12 bits
Technology	0.35um CMOS
Max Clock Frequency	330MHz
Tuning Latency	11 clock cycles
Tuning Bandwidth	165MHz (at 330MHz Fclk)
Worst-Case Spurious	-86.9 dBc
Power Dissipation	9.56mW (at 100MHz Fclk, 3.3V)
Chip Size	550 × 220 um <sup>2</sup>
Transistor Count	7,168

VI. 결 론

새롭게 제안된 직접 디지털 주파수 합성기는 기존의 DDFS에서의 사용하는 각 롬의 데이터들을 새로운 압축 알고리즘을 사용하여 양자화롭과 오차롭으로 나누어 저장한다. 제안된 DDFS의 전체 롬 크기는 기존 DDFS에서의 롬 크기에 비하여 상당히 줄어들었다. 12비트 출력 데이터를 가지는 DDFS의 경우, 롬 압축률은 78분의 1에 이른다. 성능 검증을 위하여 12비트 출력 데이터를 가지는 DDFS가 0.35um CMOS 공정으로 구현되었다. 3.3V전원과 100MHz 클럭에서 소모 전력은 9.56mW이고 최고 동작 주파수는 330MHz이다.

참 고 문 헌

[1] D. A. Sunderland, R. A. Strauch, S. S. Wharfield, H. T. Peterson, and C. R. Cole, "CMOS/SOS frequency synthesizer LSI circuit for spread spectrum communications", *IEEE JSSC*, Aug. 1984, pp. 497~505.

[2] H. T. Nicholas, H. Samuelli, and B. Kim, "The optimization of direct digital frequency synthesizer performance in the presence of finite word length effects", in *42nd Annu. Frequency Control Symp.* 1988, pp. 356~363.

[3] H. T. Nicholas, and H. Samuelli, "A 150-MHz direct digital frequency synthesizer in 1.25um CMOS with -90-dBc spurious performance", *IEEE JSSC*, Dec. 1991, pp. 1959~1969.

[4] L. K. Tan, and H. Samuelli, "A 200-MHz quadrature digital synthesizer", *IEEE JSSC*, Mar. 1995, pp. 193~200.

[5] J. Vankka, "Methods of Mapping from Phase to Sine Amplitude in Direct Digital Synthesis", *IEEE Tr. on ultrasonics, ferroelectrics, and frequency control*, Mar. 1997, pp. 526~534.

[6] J. Vankka, M. Waltari, M. Kosunen, and K. A. I. Halonen, "A Direct Digital Synthesizer with an On-Chip D/A Converter", *IEEE JSSC*, Feb. 1998, pp. 218~227.

[7] A. Bellaouar, M. S. O'brecht, A. M. Fahim, and M. I. Elmasry, "Low-Power Direct Digital Synthesis for Wireless Communication", *IEEE JSSC*, Mar. 2000, pp. 385~390.

[8] Byung-Do Yang, Ki-Hyuk Sung, Young-Joon Kim, Lee-Sup Kim, Seon-Ho Han, and Hyun-Kyu Yoo, "A Direct Digital Frequency Synthesizer Using A New ROM Compression Method", *ESSCIRC2001*, pp. 288~291.

## 저 자 소 개



梁炳燾(正會員)

1999년 2월 한국과학기술원 전자전산학과 전기및전자공학 전공 졸업 (공학사). 2001년 2월 한국과학기술원 전자전산학과 전기및전자공학 전공 졸업 (공학석사). 2001년 3월~현재 한국과학기술원 전자전산학과 전기및전자공학 전공 박사과정. <주관심분야 : 저전력/고성능 디지털 회로 설계, 저전력 메모리 설계>



金榮俊(正會員)

2000년 2월 University of Michigan 전자공학과 졸업 (공학사). 2002년 2월 한국과학기술원 전자전산학과 전기및전자공학 전공 졸업 (공학석사). 2002년 3월~하이닉스 반도체 연구원. <주관심분야 : 디지털 및 아날

로그 회로 설계>



成基赫(正會員)

1997년 2월 한국과학기술원 전자전산학과 전기및전자공학 전공 졸업 (공학사). 1999년 2월 한국과학기술원 전자전산학과 전기및전자공학 전공 졸업 (공학석사). 1999년 3월~현재 한국과학기술원 전자전산학과 전기및전자공학 전공 박사과정. <주관심분야 : VLSI 회로 설계>



金利燮(正會員)

1982년 서울대학교 전자공학과 학사. 1986년 Stanford University 전자공학과 석사. 1990년 Stanford University 전자공학과 박사. 1990년~1993년 Toshiba Corporation 연구원. 1993년~현재 한국과학기술원 전자전산학과 전기및전자공학 전공 교수. <주관심분야 : 3D Graphics 프로세서 및 하드웨어 설계, 고성능/저전력 디지털 IC 설계>