

# 위상 샘플링 방식 DRFM 적용 대역폭 제어 잡음 재밍 기법 개발

## Development of Bandwidth Controlled Noise Jamming Technique for Phase Sampling DRFM

홍상근 · 이왕용 · 류정호\* · 신욱현\*

Sang-Geun Hong · Wang-Yong Lee · Jeong-Ho Ryu\* · Wook-Hyen Shin\*

### 요 약

현대 전쟁에서는 직접 적을 타격하여 파괴하는 것뿐만 아니라 적의 전투 능력을 무력화하는 재밍의 중요성이 증가하고 있다. 잡음 재밍은 전자전 재밍 기법의 기본이 되는 기법으로 재밍하고자 하는 장비의 수신기 특성에 영향을 적게 받으며, 거의 모든 수신 장치를 교란할 수 있는 방식이다. 잡음 재밍의 경우, 출력 대비 효율을 높이기 위해 적 수신기의 대역폭과 잡음의 대역폭을 일치시킬 필요가 있으며, 다수의 서로 다른 레이더 표적을 시분할하여 재밍하기 위해 고속으로 잡음 대역폭을 가변할 필요가 있다. 본 논문은 위상 샘플링 방식으로 운용되는 DRFM에 적용할 수 있는 실시간으로 잡음 대역폭 제어가 가능한 재밍 기법을 시뮬레이션을 통해 개발하여 제안한다.

### Abstract

In modern warfare, jamming for neutralizing the enemy electronic equipments is as important as destroying them by common weapon systems. Noise jamming is a base technique of EA(Electronic Attack) and it is one of the effective jamming techniques. Noise jamming is effective regardless of enemy electronic equipment receiver types. For increasing jamming efficiency using the same output power, noise jamming bandwidth has to be similar to target receiver's bandwidth. Radar jamming source like DRFM(Digital Radio Frequency Memory) requires noise bandwidth changing immediately for time sharing multiple jamming. In this paper, we developed bandwidth changable noise jamming signal for phase sampling type DRFM and do simulation using Matlab for showing the jamming signal output.

Key words : Noise Jamming, Digital Radio Frequency Memory, Electronic Warfare, Electronic Attack

### I. 서 론

현대 전쟁에서는 직접 적을 타격하여 파괴하는 방식의 hard kill 뿐만 아니라 적의 전투 능력을 무력화하는 soft kill의 중요성이 증가하고 있다. Soft kill의 방식으로 전자전의 전자 공격인 재밍이 부각되고 있다. 재밍은 방해 전파를 송출하여 표적 수신기를 교란하는 방식으로 재밍 신호를 생성하는 소스원은

여러 종류가 있으며, DRFM(Digital Radio Frequency Memory)은 radar 재밍에 사용되는 신호원 중 하나이다.

DRFM은 동일한 위상의 신호만을 수신하는 펄스 도플러 레이더나 펄스 압축 레이더를 재밍하기 위해 반드시 필요한 재밍 장치의 핵심 구성품이다. DRFM은 입력된 신호를 메모리에 저장하였다가 필요한 시점에 저장된 신호를 복원하여 출력하는 장치이다.

(주)LIG넥스원(LIG Nex1 Co., Ltd)

\*국방과학연구소(Agency for Defense Development)

· 논문 번호 : 20110601-048

· 교신저자 : 홍상근(e-mail : hongsanggeun@lignex1.com)

· 수정완료일자 : 2011년 8월 18일

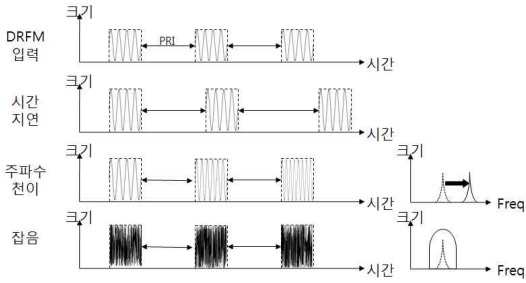


그림 1. DRFM을 이용한 재밍 개념  
Fig. 1. Jamming concept using DRFM.

DRFM에 의해 복원되어 출력된 신호는 표적 레이더의 수신단에 입력되어 표적 레이더를 교란한다. 저장된 신호를 복원하여 출력할 때 그림 1과 같이 시간 지연, 주파수 천이, 잡음 발생 등을 적용하여 여러 재밍 기법을 생성할 수 있다. DRFM은 시간 지연을 통해 RGPO/I(Range Gate Pull Off/In), 주파수 천이를 이용하여 VGPO/I(Velocity Gate Pull Off/In), 잡음발생을 이용하여 잡음 재밍, CP(Cover Pulse) 등의 재밍 기법을 구현할 수 있는 장치이다.

입력 신호를 ADC로 샘플링한 진폭 데이터 대신 I 신호(In phase)와 Q 신호(Quadrature phase)로 부터 위상 값을 찾아 저장하는 방식의 DRFM이 위상 샘플링 방식 DRFM이다.

잡음 재밍은 재밍 기법 중 기본으로 재밍하고자 하는 표적 수신기의 특성에 영향을 적게 받으며, 거의 모든 수신 장치를 교란할 수 있는 방식으로 출력 대비 효율을 높이기 위해 적 수신기의 대역폭과 잡음의 대역폭을 일치시킬 필요가 있으며, 다수의 서로 다른 레이더 표적을 시분할하여 재밍하기 위해 고속으로 잡음 대역폭을 가변할 필요가 있다.

본 논문은 위상 샘플링 방식으로 운용되는 DRFM에 적용할 수 있는 실시간으로 잡음 대역폭 제어가 가능한 재밍 기법을 시뮬레이션을 통해 개발하여 제안한다.

## II. 본 론

### 2-1 위상 샘플링 방식 DRFM 적용 방안

위상 샘플링 방식 DRFM은 입력된 고주파 신호들의 위상 값을 저장하며, 필요한 시점에 저장된 위상 데이터를 재밍 기법에 따라 가공하여 출력하는 재밍

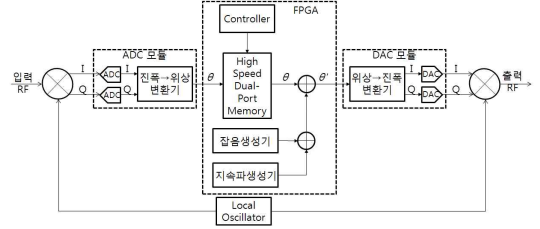


그림 2. DRFM 블록도  
Fig. 2. DRFM block diagram.

소스원이다<sup>[5]</sup>. 위상 샘플링 방식 DRFM은 입력 RF 신호를 I 신호와 Q 신호로 변환한 후 ADC를 이용하여 샘플링한다. 이렇게 샘플링된 데이터로부터 입력 RF신호의 위상 값을 산출하여 메모리에 저장한다. 저장된 신호를 출력할 때 저장된 위상 값으로부터 I 신호와 Q 신호의 진폭 데이터로 변환되어 DAC로 출력된다<sup>[1]</sup>. 위상 샘플링 방식 DRFM에서 위상 값의 산술 연산을 통해 VGPO/I, 잡음 재밍, CP 등의 재밍 기법을 구현할 수 있다. 그림 2는 위상 샘플링 방식 DRFM의 계통도를 간략히 도시한 것이다.

위상 샘플링 방식 DRFM에 사용되는 ADC 모듈은 동기화된 2채널의 ADC와 진폭을 위상으로 변환하는 기능을 가지며, DAC 모듈은 위상을 진폭으로 변환하는 기능과 동기화된 2채널의 DAC를 가진다. I 신호와 Q 신호에 따른 위상값( $\theta$ )은 식 (1)과 같으며, 위상값( $\theta$ )에 따른 I 신호와 Q 신호의 진폭은 식 (2), 식 (3)과 같다. 여기서  $A$ 는 출력 신호의 크기를 의미한다.

$$\theta = \arctan \frac{Q}{I} \tag{1}$$

$$I = A \cos(2\pi\theta) \tag{2}$$

$$Q = A \sin(2\pi\theta) \tag{3}$$

입력의 대역폭을 갖는 잡음 신호를 저장된 신호의 주파수에 실어 출력하는 것이 DRFM을 이용한 잡음 재밍 방안이다. 만약 저장된 고주파 신호의 위상 데이터가 없더라도 DRFM은 지속파를 생성하는 기능을 가지고 있어 지속파에 잡음을 실어 출력할 수도 있다. 위상 샘플링 방식 DRFM의 경우 저장된 데이터가 위상 값이며, DAC 모듈 전단까지 위상 정보만을 이용하므로 잡음을 생성하는 연산 과정 또한 위상값을 이용한다.

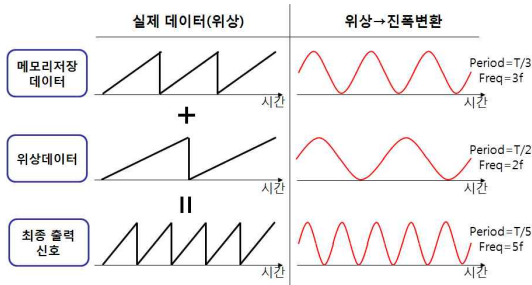


그림 3. 위상 산술 연산 알고리즘의 예  
Fig. 3. Example of phase arithmetic algorithm.

그림 2에서 잡음 생성기나 지속과 생성기를 통해 생성된 위상 데이터는 메모리에 저장된 데이터와 산술 연산을 통해 재밍 기법이 구현된다. 두 개의 위상 데이터를 산술 연산할 경우 그림 3과 같이 두 위상 데이터가 나타내는 주파수를 더하는 것과 같다.

지속과 생성기의 출력 위상 데이터와 메모리에 저장된 데이터를 산술 연산하면 메모리에 저장된 신호의 주파수가 지속과 생성기의 출력 위상 데이터의 주파수만큼 천이되는 효과가 있다. 잡음 생성기에서 출력된 잡음 위상 데이터를 메모리에 저장된 데이터와 산술 연산을 하면 메모리에 저장된 데이터 신호의 주파수를 중심으로 잡음 재밍 신호가 출력된다.

### 2-2 대역 제어 잡음 구현 방안

잡음 대역폭(NBW: Noise BandWidth)을 제어하여 생성된 잡음 생성기의 위상 잡음 데이터는 메모리에 저장된 데이터와 산술 연산을 하여 메모리에 저장된 데이터의 중심 주파수로 천이하게 된다. 그러므로 잡음 생성기의 위상 잡음 데이터는 DAC를 통해 최종 출력 시 주파수 범위가 메모리에 저장된 데이터의 중심 주파수를 기준으로  $-NBW/2$ 에서  $NBW/2$

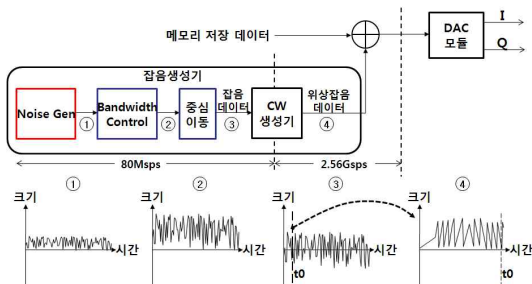


그림 4. 잡음 생성기 개념도  
Fig. 4. Noise generator concept.

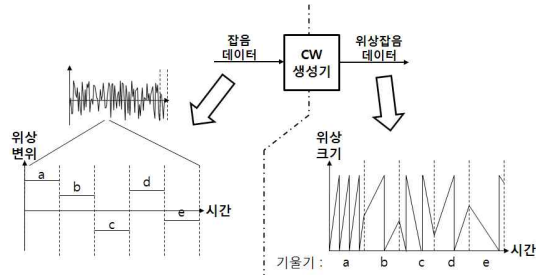


그림 5. CW 생성기 개념도  
Fig. 5. CW generator concept.

사이의 값을 갖는다.

그림 4는 잡음 생성기의 블록 다이어그램으로 잡음생성 방안을 간략히 나타낸 것이다. 디지털로 1 kHz의 잡음을 생성①하고 대역폭을 곱셈기를 이용하여 1 kHz 단위로 제어한다②. 대역폭이 제어된 잡음의 값이 0에서 NBW 사이의 값을 갖고 있는데, 이를  $-NBW/2$ 에서  $NBW/2$ 의 값을 갖도록 중심 이동한다③. 중심 이동된 잡음 데이터는 위상 변위로 그림 4의 CW 생성기에서 위상 값으로 변환되어 위상 잡음 데이터로 출력된다④. 여기서 CW 생성기의 기본 개념은 그림 5와 같다. 최종적으로 산술 연산을 통해 위상 잡음 데이터는 메모리에 저장된 데이터의 주파수를 중심으로 출력된다.

1 kHz의 대역폭을 가지는 기본 단위 잡음을 생성하기 위해 디지털로 32비트 랜덤 신호를 생성하여 잡음 소스원으로 사용한다. 잡음 소스원으로 PRN (Pseudo Random Noise)를 생성하기 위해 LFSR(Linear Feedback Shift Register)를 이용하여 Alternating step generator 형태를 구성하였다. Alternating step generator는 세 개의 LFSR을 이용하여 한 개의 LFSR을 사용하였을 때 반복 주기가 짧아지는 문제를 해결하였다. 그림 6은 alternating step generator를 이용한 pseudo random noise 생성 블록의 개념도로 LFSR1의 출력수열이 '1'이면 LFSR2를 동작시키고, '0'이면 LFSR3를 동작시켜 LFSR2의 출력과 LFSR3의 출력을 XOR하여 최종 출력을 결정한다<sup>[2]</sup>.

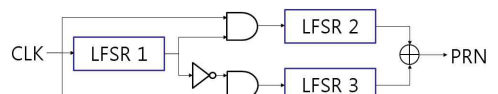


그림 6. PRN 생성부 개념도  
Fig. 6. PRN generator concept.

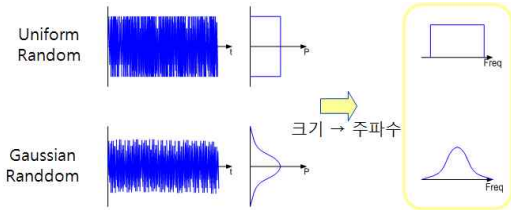


그림 7. Uniform과 Gaussian 분포 비교  
Fig. 7. Uniform vs Gaussian.

그림 6의 PRN 생성부의 반복 주기는 alternating step generator의 최대 반복 주기와 같다. PRN 생성부를 구성하는 각각의 LFSR 비트수가  $n_1, n_2, n_3$ 이고, 각각의 LFSR이 최대 주기를 가진다면 반복 주기(L)는 식 (4)와 같다(gcd: 최대 공약수).

$$L = (2^{n_1} - 1) \cdot \text{gcd}(2^{n_2} - 1, 2^{n_3} - 1) \quad (4)$$

그림 7은 균일(uniform) 분포와 가우시안(Gaussian) 분포의 차이에 의한 위상 샘플링 방식 DRFM의 최종 출력의 주파수 특성을 비교한 그래프이다. PRN 생성부는 균일 분포를 갖는 PRN 신호를 출력한다. PRN 생성부에서 생성된 잡음을 위상 샘플링 방식 DRFM에서 실제 주파수 축으로 변환되므로 PRN 생성기에서 가우시안 분포를 갖는 신호를 이용하는 것보다 균일 분포를 갖는 것이 대역 내 평탄도면에서 유리하다.

PRN 생성부의 출력 신호에 1 kHz에 해당하는 주파수 값을 곱하여 1 kHz의 대역폭을 가지는 잡음 데이터를 생성한다.

임의의 잡음 대역폭을 가지는 잡음 신호를 만들기 위해 1 kHz의 대역폭을 가지는 PRN 생성부의 출력 신호에 kHz 단위의 대역폭 값을 곱하여 원하는 대역폭의 잡음 신호를 추출한다. 이 때 잡음 대역폭은 0~NBW 사이의 값을 갖는다.

잡음 신호를 중심 이동하기 위해 대역폭이 제어된 결과에 잡음 대역폭의 1/2만큼 음의 방향으로 offset을 주어  $-NBW/2 \sim NBW/2$  사이 값을 갖는 잡음 데이터를 생성한다.

대역폭 제어와 중심이동이 된 잡음 데이터는 단위 시간 당 위상 변위 값으로 이를 위상으로 변환하는 과정을 CW 생성기에서 수행한다. CW 생성기는 입력으로 위상 변위( $\Delta p$ )를 받아 위상 값을 출력하

여 DAC가 출력하는 신호의 실제 주파수를 생성되도록 한다. 즉, 생성된 잡음 신호의 값이 CW 생성기에서 위상값으로 변경되어 DAC의 출력에서 주파수로 변환된다. CW 생성기의 입력 값, 위상 변위( $\Delta p$ )와 DAC 출력 주파수( $F$ )와의 관계는 식 (5)와 같다. 여기서  $F_{clk}$ 는 DAC의 샘플링 클럭 주파수이며,  $B$ 는 DAC의 입력 비트수이다.

$$F = \frac{\Delta p \cdot F_{clk}}{2^B} \quad (-2^{B-1} \leq \Delta p \leq 2^{B-1} - 1) \quad (5)$$

일반적으로 DRFM에 사용되는 ADC, DAC는 높은 샘플링 주파수를 가진다. ADC, DAC와 같이 높은 샘플링 주파수를 갖는 클럭을 데이터를 처리하는데 사용할 수 없다. 그러므로 CW 생성기는 낮은 주파수 클럭을 이용하여 DAC의 여러 샘플링 데이터를 동시에 처리한다.

CW 생성기라 명칭하는 이유는 DDS(Direct Digital Synthesizer)와 유사하게 CW 생성기는 입력 값을 주파수로 인식하여 출력을 정현파를 생성하는 것과 유사하기 때문이다. CW 생성기에 의해 DAC 입력에 맞도록 위상 값으로 변환된 위상 잡음 데이터는 메모리에 저장된 신호와 각 샘플 데이터마다 더하여 출력된다. 이 과정을 통해 최종 잡음 재밍 신호의 중심 주파수가 메모리에 저장된 신호의 주파수가 된다. 이 때 잡음 신호와 메모리에 저장된 신호의 연산 결과 위상 값이  $2\pi$ 를 넘어서  $2\pi + \theta$ 의 값일 경우  $2\pi$ 를 버리고  $\theta$ 만을 취한다. 그림 8은 5개의 DAC 샘플링 데이터를 동시에 처리하는 CW 생성기의 예시이다.

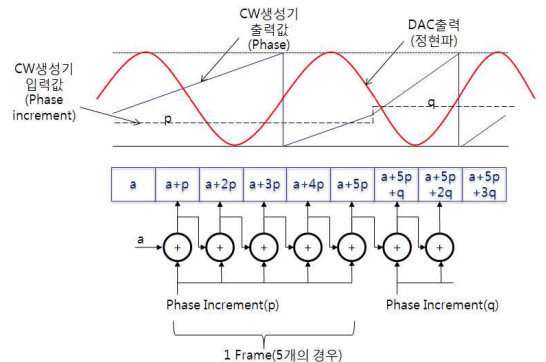


그림 8. CW 생성 알고리즘 예시  
Fig. 8. Example of CW generator algorithm.

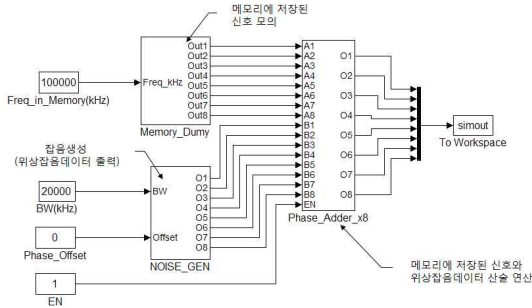


그림 9. 시뮬레이션 블록도  
Fig. 9. Simulation block diagram.

2-3 대역폭 제어 잡음 생성 시뮬레이션 구성

대역 제한 잡음 생성 방안의 타당성을 확인하기 위해 시뮬레이션을 matlab simulink를 이용하여 수행하였다. 시뮬레이션은 pseudo random code를 이용하여 임의의 대역폭을 가지는 잡음 신호의 생성부터 DAC 입력 데이터 형태로 변환하는 것까지 수행하였다. 시뮬레이션 결과데이터를 I/Q mixer를 통과하여 실제 출력 결과와 같도록 모의하였다.

그림 9는 시뮬레이션의 블록도이다. 시뮬레이션을 위해 대역폭 제어가 가능한 잡음 신호 생성부(NOISE\_GEN)와 산술 연산(Phase\_Adder\_x8)을 수행하는 부분, 그리고 메모리에 저장된 데이터를 모의해주는 부분(Memory\_Dummy)으로 구성된다.

그림 10은 잡음 신호를 생성하는 NOISE\_GEN 블록의 상세 블록도이다. 그 중 PRN 신호 생성하는 블록(Noise\_Gen\_32 bit)의 상세 구조는 그림 11과 같다.

그림 11의 PRN 신호 생성 블록(Noise\_Gen\_32 bit)은 그림 6의 alternating step generator를 그대로 적용하였으며, XOR 대신 switch를 사용하였다. 여기서

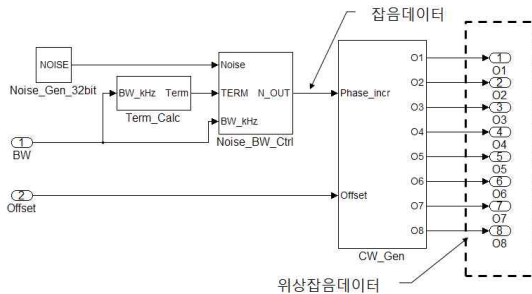


그림 10. 잡음생성부 블록도  
Fig. 10. Noise generator block diagram.

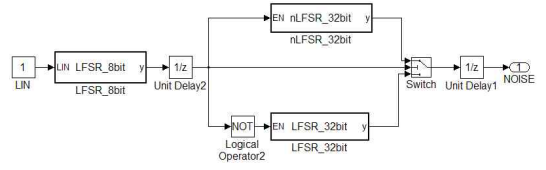


그림 11. PRN 생성부 블록도  
Fig. 11. PRN generator block diagram.

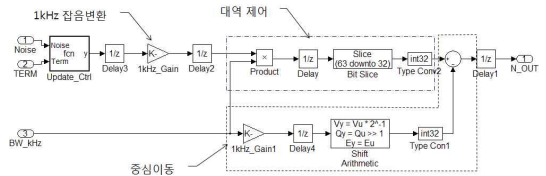


그림 12. 잡음 대역 제어 및 중심이동 블록도  
Fig. 12. Noise BW control & center moving block diagram.

실제 잡음을 생성하는 LFSR(LFSR\_32 bit, nLFSR\_32 bit)은 32 bit로 동작하며, switch를 제어하고 LFSR을 enable시키는 신호는 8 bit LFSR(LFSR\_8 bit)을 이용하여 구현하였다. 이 경우 최대 반복 주기는 80 MHz 클럭으로 동작을 가정할 때 약 3시간 48분의 주기를 가지므로 반복성을 찾기 어렵다.

그림 12는 잡음 신호 대역폭을 제어하고 중심을 이동하는 기능을 수행하는 블록으로 그림 10의 Noise\_BW\_ctrl 블록의 상세 블록도이다. 초단에서 입력된 32 bit PRN 신호에 1 kHz에 대응되는 상수를 곱셈하여 1 kHz를 만든다. 곱셈 결과 bit 수가 증가하는데 64 bit로 확장하여 수행하였다. 이는 fixed point 연산을 정수부 32 bit 소수부 32 bit로 운용하는 것과 같다. 이렇게 생성된 1 kHz 잡음 신호에 곱셈 연산을 하여 kHz 단위로 잡음 대역폭의 제어한다. 대역폭 제어가 완료된 잡음 신호의 정수부 32 bit만을 취하여 중심 이동을 처리한다. 중심 이동은 제어한 대역폭의 절반만큼 이동을 시킨다. 입력된 대역폭 제어 값은 kHz 단위이므로 실제 데이터의 형태에 맞게 값을 변환한다. 대역폭의 절반 값을 변환된 대역폭 값을 사칙 연산 대신 1 bit right shift를 수행하여 생성한다. 이렇게 생성된 대역폭의 절반 값을 최종 잡음 출력에서 뺄셈 연산을 통해 중심 이동이 구현된다.

그림 8의 CW 생성 알고리즘을 적용한 CW 생성기의 블록도는 그림 13이다. CW 생성기의 입력 값

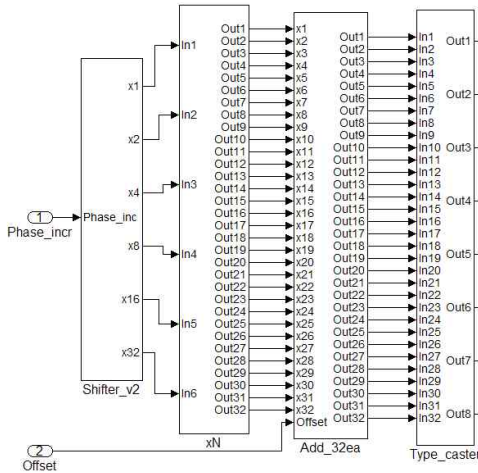


그림 13. CW 생성기 블록도  
Fig. 13. CW generator block diagram.

이 DAC 출력의 주파수가 되도록 DAC 입력 값을 생성하는 블록이다. CW 생성기는 32개의 샘플을 동시에 처리한다. 즉, 입력 값의 1배수부터 32배수까지 동시에 구하여 연산을 수행한다. DAC의 클럭과 디지털 처리하는 클럭의 주파수에 차이가 있다. CW 생성기는 DAC 입력 데이터를 만드는데 반해 디지털 처리를 위한 상대적으로 낮은 주파수의 클럭을 사용한다. 그러므로 CW 생성기는 1 클럭에 다수의 신호를 동시에 처리해야 한다. CW 생성기는 과거의 최종 값에 입력을 1배부터 32배한 것을 각각 더하여 출력하고, 출력 중 32배한 것과 과거 값을 더한 값이 다음 클럭에서 더하는 과거의 최종 값이 된다.

입력 값의 1배부터 32배 한 값을 1 클럭에 연산하기 위해 32개의 곱셈기를 사용하지 않고, shift 연산과 덧셈의 조합으로 구현한다. 1, 2, 4, 8, 16, 32배수 값은 shift 연산으로 전처리를 수행한다. 전처리된 값을 조합하여 1부터 32배의 값을 계산해 과거의 최종 값에 동시에 더하여 출력한다. 예로 9배수의 값을 구할 때 입력을 3 bit left shift하여 8배수를 만들어 1배수 값과 더하여 9배수 값을 구한 후 과거의 최종 값에 더한다.

그림 14는 메모리에 저장된 신호와 산술 연산을 수행하는 블록으로 그림 9의 산술 연산기(Phase\_Adder\_x8) 블록도이다. 산술 연산기는 메모리에 저장된 32개의 샘플 데이터와 잡음 데이터를 동

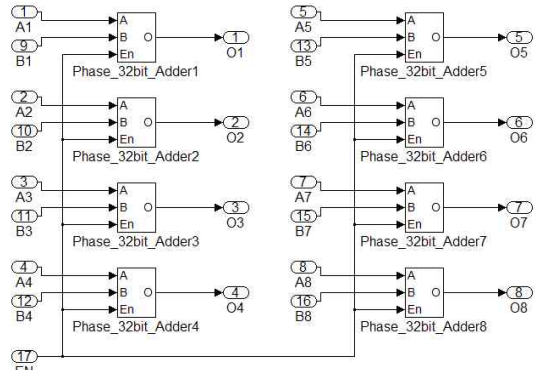


그림 14. 산술 연산기 블록도  
Fig. 14. Arithmetic block diagram.

시에 더하고 연산결과를 DAC의 입력 비트 수에 맞게 변환하여 출력한다.

#### 2-4 대역폭 제어 잡음 생성 시뮬레이션 결과

시뮬레이션은 메모리에 저장된 신호의 주파수가 100 MHz일 때 20 MHz 대역폭을 갖는 잡음을 출력하는 경우를 수행하였다. 최종 DAC 출력은 I/Q 각각이 출력되는데 이를 I/Q mixer를 통해 2.5 GHz로 상향 변환하는 경우로 제한한다. 이 경우 메모리에 저장된 100 MHz 신호에 의해 2.6 GHz를 중심으로 하여 20 MHz의 대역폭을 갖는 잡음 재밍 신호가 출력

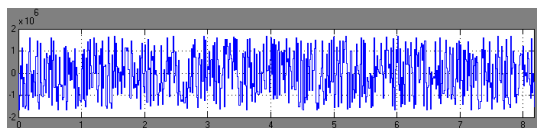


그림 15. 잡음 데이터  
Fig. 15. Noise data.

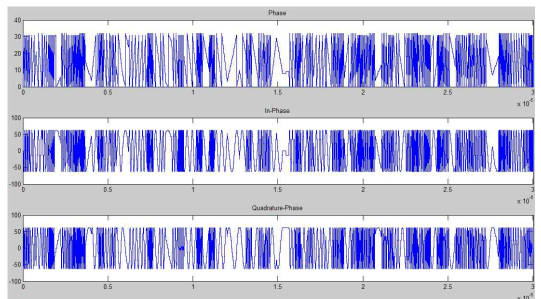


그림 16. DAC 입 · 출력  
Fig. 16. DAC input & output.

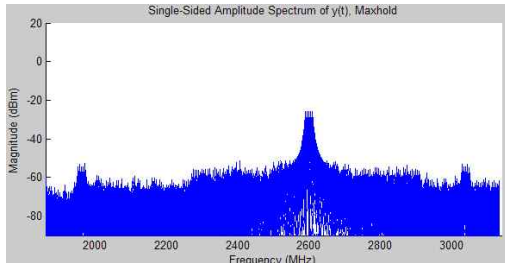


그림 17. DRFM 최종 출력  
Fig. 17. DRFM output.

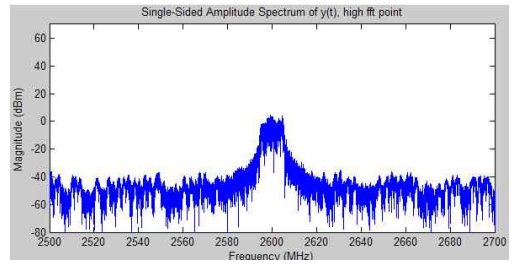


그림 20. DRFM 최종 출력(대역폭 10 MHz)  
Fig. 20. DRFM output(bandwidth 10 MHz).

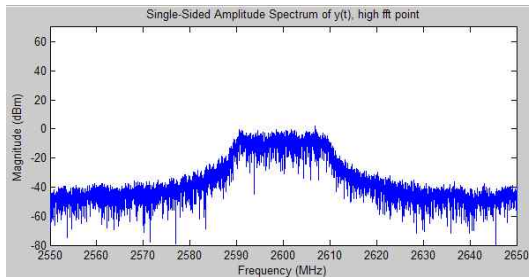


그림 18. DRFM 최종 출력(확대)  
Fig. 18. DRFM output(zoom).

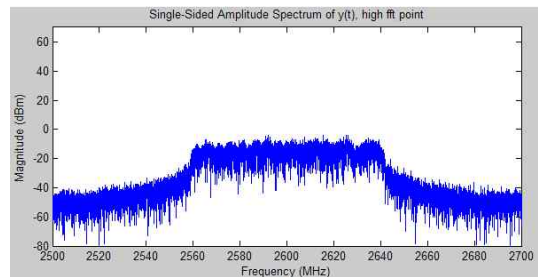


그림 21. DRFM 최종 출력(대역폭 80 MHz)  
Fig. 21. DRFM output(bandwidth 80 MHz).

된다.

그림 15는 잡음 대역폭이 20 MHz가 되도록 제어된 pseudo random noise를 시간축 상에서 본 것이다. 이 데이터를 가지고 위상 잡음 데이터로 변환하여 DAC의 입·출력을 확인한 것이 그림 16이다. 그림 16의 첫 번째 그래프는 DAC의 입력이 되는 위상 값이다. 두 번째와 세 번째 그래프는 DAC 출력인 I 신호와 Q 신호의 출력이다. DAC의 출력 신호를 I/Q mixer를 이용하여 2.5 GHz의 LO로 상향 변환한 결과가 그림 17이다. 메모리에 저장된 신호의 주파수가 100 MHz이므로 2,600 MHz의 중심 주파수를 가지는 잡음 신호가 출력된다. 그림 18은 그림 17의 잡음

음 출력을 확대하여 나타낸 것이다.

잡음 대역폭은 그림 10의 BW 입력을 1 kHz 단위로 입력하여 제어한다. 실시간으로 잡음 대역폭이 제어되는지 확인하기 위해 10 MHz와 80 MHz를 교번하여 잡음 신호를 생성하였다. 잡음 대역폭은 1 kHz 단위로 제어가 가능하지만 변화를 알기 어려우므로 10 MHz와 80 MHz로 시험을 수행하였다. 실시간으로 잡음 대역폭이 제어되는 그림 19를 통해 확인하였다. 그림 19의 첫 번째 그래프는 입력 잡음 대역폭이고, 두 번째 그래프는 잡음 데이터이다. 세 번째 그래프는 잡음 데이터가 CW\_GEN을 통과한 위상 잡음 데이터이며, 네 번째 그래프는 최종 출력이다. 10 MHz와 80 MHz 잡음의 최종 출력을 FFT 수행한 것은 그림 20, 21이다.

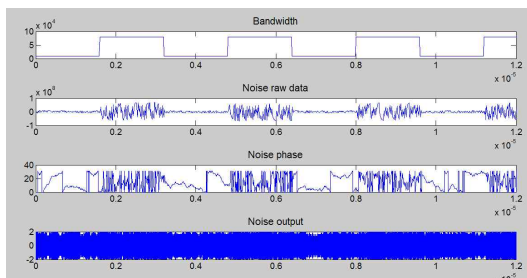


그림 19. 잡음 대역폭 제어 확인  
Fig. 19. Controlled noise BW.

### III. 결 론

본 논문에서 위상 샘플링 방식의 DRFM에 적용하기 위한 대역폭 제어가 가능한 잡음 재밍 기법의 구현방안을 제시하였다. 제시한 방안은 실제로 FPGA를 이용하여 구현될 것을 고려하여 FPGA에서 적용이 용이하도록 하였다. FPGA에서 적용이 쉬운 shift

연산과 가감산을 대부분 사용하고 필요한 부분에서만 곱셈기를 사용하였다. 다중 재밍 시 수 클럭 이내에 잡음 대역폭을 변경할 수 있도록 하였다. 본 논문에서 제시한 잡음 재밍 기법 구현 방안은 위상 샘플링 방식 DRFM에서 FPGA의 상대적으로 느린 클럭을 이용하여 고속의 DAC를 제어하며, 실시간으로 대역폭 제어가 가능한 잡음 재밍 신호를 생성하여 시분할 다중 재밍에 적용할 수 있다.

### 참 고 문 헌

- [1] 유병석, "디지털 고주파 메모리 구현에 관한 연구", 한국해양정보통신학회논문지, 14(9), pp. 2164-2170, 2010년.
- [2] Erik Zenner, "On the efficiency of the clock control guessing attack", *ICISC*, 2002.
- [3] C. G. Günther, "Alternating step generators controlled by de Bruijn sequences", *Advances in Cryptology-EuroCrypt '87*, pp. 5-14, 1987.
- [4] James B. Dabney, Thomas L. Harman, *Mastering SIMULINK4*, Prentice Hall, 2001.
- [5] Phillip E. Pace, *Advanced Techniques for Digital Receivers*, Artech House, 2000.
- [1] 유병석, "디지털 고주파 메모리 구현에 관한 연구"

### 홍 상 근



밍 기법 M&S

2004년 2월: 고려대학교 전자공학과 (공학사)  
 2010년 5월: 성균관대학교 전자공학과 (공학석사)  
 2004년 1월~현재: LIG넥스원 전자전연구센터 선임연구원  
 [주 관심분야] 전자전 EA 시스템, 재

### 류 정 호



Embedded System 개발

1990년 2월: 청주대학교 전자공학과 (공학사)  
 1992년 2월: 청주대학교 전자공학과 (공학석사)  
 1992년 3월~현재: 국방과학연구소 선임연구원  
 [주 관심분야] 전자전 EA 시스템, Em-

### 이 왕 용



호정보수집/재밍시스템

1993년 2월: 경북산업대학교 전자공학과 (공학사)  
 2007년 2월: 아주대학교 정보전자공학과 (공학석사)  
 1995년 8월~현재: LIG넥스원 전자전연구센터 수석연구원  
 [주 관심분야] 전자전 시스템, 신

### 신 옥 현



책임연구원

[주 관심분야] 전자전 시스템, EA 시스템

1983년 2월: 경북대학교 전자공학과 (공학사)  
 1986년 2월: 경북대학교 전자공학과 (공학석사)  
 2003년 8월: 충남대학교 컴퓨터학과 (이학박사)  
 1986년 2월~현재: 국방과학연구소