

DRNS용 SRTFR 변환기 설계에 관한 연구

김 용 성*

목 차

- I. 서 론
- II. DRNS를 사용한 연산구조
- III. CRT를 사용한 DRNS용 RTB변환기 설계
- IV. 3D구조의 혼합기수변환(MRC)을 사용한 DRNS용 RTB변환기 설계
- V. 2D구조의 MRC 및 CRT를 사용한 DRNS용 RTB변환기 설계
- VI. 실험 및 고찰
- VII. 결론.

I. 서 론

잉여수계(RNS: Residue Number System)는 비가중치 수 체계(Unweighted Number System)로 모듈리(moduli)간에 캐리 전달이 필요 없고, 병렬로 연산이 처리되므로, 신경망 회로, 그래픽, 디지털 영상처리 등과 같은 여러 응용분야에서 고속의 전용연산에 적합한 구조로 사용되어 왔다. [1] [2] [3] [4] [5]

잉여수계는 각 모듈리 별로 수를 분할하여 연산

을 수행하므로 적은 크기의 연산기로 연산이 가능하다.[6] DRNS(Double Residue Number System)는 다중 잉여수 변환 처리 방법 중 효율적인 방법의 하나로서, 이중으로 잉여수로 변환하여 연산처리를 함으로써 연산기의 크기를 감소시키는 장점이 있다.[7] 잉여수계 연산기 중에서 1 Out-of-m코드를 순환부호로 사용한 연산기는 고속의 연산처리가 가능하지만 모듈리가 소수(prime number)인 경우에만 연산이 가능하다는 제한 점을 갖고, 베릴 쉬프터를 기본구조로 사용하여 연산기가 커지는 문제점이 있다.[8] 이와 같은 연산기를

* 여주대학 게임엔터테인먼트과 교수

사용하는 경우 DRNS를 사용하면 1차 잉여수 변환에서는 서로소인 모듈리를 사용하고 2차 잉여수에서는 소수인 모듈리를 사용함으로써 이러한 제한점을 보완할 수 있으며, 연산기의 크기도 감소시킬 수 있다. 그러나 잉여수를 기중치 수체계인 2진수로 변환하는 변환기(RTB: Residue to Binary Converter) 경우 2번의 변환과정이 필요하게 되므로, 특정한 응용범위에서 대한 모듈리를 선택하는 경우를 제외하면 연산기의 크기가 증가하는 문제점을 갖는다. CRT(Chinese Remainder Theorem)를 사용한 고속화 방법은 $\{2^n - 1, 2^n, 2^n + 1\}$ 과 같이 한정된 모듈리에 적용하는 경우가 대부분이므로, [9] [10] 소수 만을 사용하는 경우나 연산기 크기를 고려하여 모듈리를 작게 선택하는 경우에는 적합하지 않다. 그러므로 본 논문에서는 SRTFR (Second Residue to First Residue) 변환기 설계 시 연산기의 크기를 감소시키기 위하여 혼합기수계 (MRC: Mixed Radix Conversion)을 사용한 변환기를 설계하고 이를 사용하여 연산기의 크기를 감소된 RTB 변환기를 설계하고자 한다.

II. DRNS를 사용한 연산구조

잉여수 연산에서 모듈리(moduli)의 개수 및 수의 범위는 연산에 사용되는 수의 범위에 따라 정해지게 되는데, 사용될 수의 범위를 E라 하고 서로 소인 모듈리 P에 대한 잉여수 표현은 다음과 같이 표현할 때, 연산에 상되는 수의 범위는 식(1)과 같이 표현된다. [6]

$$P = \{m_1, m_2, m_3, \dots, m_n\},$$

$$R = \{r_1, r_2, r_3, \dots, r_n\} \quad (n: \text{정수})$$

$$M \geq E, \quad (M = \prod_{i=1}^n m_i) \quad (1)$$

정수 X, Y 두수 이항연산('o')의 결과를 Z라고 하는 경우, $Z = X \circ Y$ 연산에 대한 모듈러스 m_i

에 대한 잉여수 연산은 식(2)와 같이 표현된다.

$$|Z|_{m_i} = |X \circ Y|_{m_i} = (X \circ Y) \bmod m_i \quad (\text{단, 'o' 는 } +, -, \times) \quad (2)$$

제산의 경우는 곱의 역을 기본으로 사용하며, 스케일링(Scaling)과 일반적인 제산법을 사용하며, 일반적인 제산은 근사화된 제수를 사용하며, 연산결과에 따른 피제수와의 오차를 구하여 이를 보정하는 방법을 주로 사용한다. [11] [12] 모듈러스 m_i 에 대한 정수 a의 곱의 역을 식(3)에 표시하였다.

$$|ad|_{m_i} = |1|_{m_i}, \quad |d|_{m_i} = |1//a|_{m_i} \quad (\text{단, } a, m_i \text{는 서로소이며 정수}) \quad (3)$$

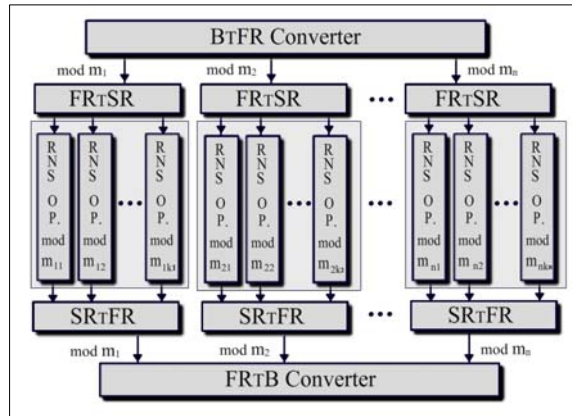


그림 1. DRNS를 사용한 연산 구조

DRNS는 1차 잉여수 변환된 결과에 다시 2차 잉여수로 변환하는 것이므로, 1차 잉여수의 모듈러스 m_i 에 대한 2차 잉여수는 식(4)와 같이 표현되며, 2차 잉여수의 연산범위 포함할 수 있도록 정해져야 한다.

$$P_i = \{m_{i1}, m_{i2}, m_{i3}, \dots, m_{ikn}\}, \quad m_i \leq M_2,$$

$$M_2 = \prod_{i=1}^{kn} m_{ij} \quad (i, kn: \text{정수}) \quad (4)$$

예를 들어 1차 잉여수의 모듈러스가 15이며 두수의 잉여수 승산인 경우는 연산 범위가 196보다 큰 수이어야 하므로, 2차 잉여수의 모듈리는 2, 3, 5, 7로 선택하며, 연산 결과와 합이 필요한 경우는 3, 4, 5, 7을 선택한다

그림 1에 DRNS를 사용한 연산 구조를 나타내었

다. 입력된 2진 데이터는 BTFR(Binary to First Residue)변환기를 통하여 식(3)에 따라 mod m_1 에서 mod m_n 까지 1차 잉여수로 변환되고, 변환된 데이터는 FRTSTR(First Residue to Second Residue)변환기를 통하여 각 모듈러스마다 식 (3)에 따라 mod m_{11} 에서 mod m_{nk_n} 까지 2차 잉여수로 전환되어 잉여수 연산이 병렬로 수행되며, 연산된 결과는 SRTFR(Second Residue to First Residue) 변환기와 FRTB(First Residue to Binary)로 구성된 RTB(Residue to Binary)통하여 2차 잉여수에서 1차 잉여수로 변환되고, 다시 2진수 데이터로 변환되어 출력한다. 2진 데이터를 잉여수로 변환하는 변환기는 2진 데이터와 잉여수 표현과의 대응 표에 따라 인코더를 설계하여 사용한다. SRTFR과 FRTB 변환기는 일반적으로 CRT(Chinese Remainder Theorem)을 2번사용하여 설계하는데 연산기의 크기가 증대하는 요인이 된다. 그러므로 본 논문에서 SRTFR 변환기에 혼합기수변환(MRC: Mixed Radix Conversion)을 사용하여 효율적인 RTB 변환기를 설계하고자 한다.

III. CRT를 사용한 DRNS용 RTB변환기 설계

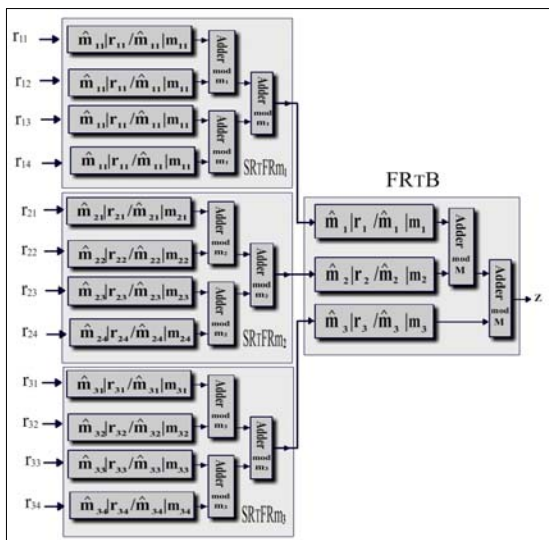


그림 2. CRT를 사용한 DRNS용 RTB변환기(n=3, kn=4)

CRT는 잉여수를 이진수로 변환하는 일반적인 방법으로 모듈리가 $\{m_1, \dots, m_n\}$ 까지 n개이고, 각 모듈리 별 잉여수가 $\{r_1, \dots, r_n\}$ 일 때, 변환 식은 식(5)와 같다.

$$|z|_M = \left| \sum_{i=1}^n \widehat{m}_i |r_i| / \widehat{m}_i |m_i| \right|_M, \quad (\widehat{m}_i = \prod_{h=1, h \neq i}^n m_h, \quad (5))$$

DRNS에서 I번째 1차 잉여수 m_i 에 대한 2차 잉여수의 모듈리가 k_i (정수)개일 때, 모듈리 $P_i = \{m_{i1}, \dots, m_{i k_i}\}$ 라 하면, 이에 대한 2차 잉여수 표현은 $R = \{r_{i1}, r_{i2}, r_{i3}, \dots, r_{i k_i}\}$ 이 된다.

다. 이 경우 SRTFR변환을 위한 CRT표현은 식(6)과 같다.

$$|r_i|_{m_i} = \left| \sum_{j=1}^{k_i} \widehat{m}_{ij} |r_{ij}| / \widehat{m}_{ij} |m_{ij}| \right|_{M_2 |m_i}, \quad (\widehat{m}_{ij} = \prod_{g=1, g \neq j}^{k_i} m_{ig}, \quad \text{단 } g \neq j) \quad (6)$$

RTB 변환기는 식 (5)와 식(6)에 의하여 수행된다. 1차 잉여수의 모듈리가 3개(n=3)이고, 2차 잉여수의 모듈리가 4개(k1=k2=k3=4)인 경우, CRT를 사용한 DRNS용 RTB 변환기를 그림 2에 나타내었다. 그림 2에서 DRNS 잉여수 입력 $\{r_{i1}, r_{i2}, r_{i3}, r_{i3}\}$ 에 대한 1차 RNS 모듈리 m_i 에 대한 변환을 SRTFR m_i 에서 수행한다.

식 (6)에서 $\sum_{j=1}^{k_i} \widehat{m}_{ij} |r_{ij}| / \widehat{m}_{ij} |m_{ij}| = B$ 라 할 때, B는 식 (7-1)의 범위를 가지므로, 식(7-2)와 같이 모듈리의 교환이 성립되지 않으므로 B연산 시 연산기의 크기가 감소되지 않는 단점을 갖는다.

$$0 \leq B < nM_2 - \sum_{j=1}^{k_i} \widehat{m}_{ij}, \quad (7-1)$$

$$\|B\|M_2|m_i \neq \|B\|m_i|M_2 \quad (7-2)$$

그러므로 SRTFR_{m_i}의 변환은 식(8)과 같이 연산과정을 두 부분으로 분리하고 각각의 잉여수 가산기를 적용하여 최종적으로 이에 대한 잉여수 가산을 수행한다.

$$C = |\widehat{m}_{i1}|r_{i1} // \widehat{m}_{i1}|m_{i1} + |\widehat{m}_{i1}|r_{i1} // \widehat{m}_{i1}|m_{i1}|M_2$$

$$D = |\widehat{m}_{i1}|r_{i1} // \widehat{m}_{i1}|m_{i1} + |\widehat{m}_{i1}|r_{i1} // \widehat{m}_{i1}|m_{i1}|M_2$$

$$|r_i|m_i = \|(C + D)M_2|m_{i1} \quad (8)$$

SRTFR_{m_i}에 의하여 변환된 결과는 FRTB에서 식(6)의 연산을 수행하여 잉여수를 2진수로 변환한다. LUT(Look-up Table)만을 사용하여 설계하는 경우에는 그림 2.와 설계하는 경우 LUT의 크기가 증대하게 되므로, SRTFR_{m_i}에서 r_{i1} ~ r_{ki}의 입력으로 하여 하나의 LUT를 사용하여 설계하면 $(\prod_{j=1}^{k_i} m_{ij})(\lceil \log_2 m_i - 1 \rceil + 1)$ 비트의 크기로 설계할 수 있다.

IV. 3D구조의 합기수변환(MRC)을 사용한 DRNS용 RTB변환기 설계

혼합기수변환(MRC: Mixed Radix Conversion)은 비가중치 수체계를 가중치 수체계로 변환하는 방법으로 잉여수의 대소 및 부호 판별 등에 주로 사용되지만, 본 논문에서는 MRC를 사용하여 RTB 변환기 설계에 적용하고자 한다.

정수 Y에 대한 혼합기수 표현이 {a₁, a₂, ..., a_n}인 경우, 모듈리 m_i의 곱 {m₁, ..., $\prod_{i=1}^{n-1} m_i$ }을 혼합기수로 사용하고, 계수를 a_n으로 표시하면, 임의 정수 Z의 혼합기수 표현은 식(9)와 같다.

$$Z = a_n \prod_{i=1}^{n-1} m_i + \dots + a_3 m_1 m_2 + a_2 m_1 + a_1 \quad (9)$$

식(9)에서 |Z|m₁ = a₁이므로 a₁ = r₁이 되고, 다음 식과 같이 표현하면 a₂, a₃는 식(10), 식(11)과 같이 표현할 수 있다.

$$(Z - a_1)/m_1 = (a_n \prod_{i=3}^{n-1} m_i + \dots + a_3)m_2 + a_2$$

$$|a_2|m_2 = |(Z - a_1)/m_1|m_2 \quad (10)$$

$$|a_3|m_3 = |((Z - a_1)/m_1 - a_2)/m_2|m_3 \quad (11)$$

위와 동일한 방법을 순환식으로 표시하면, 식(12)와 같이 표시할 수 있다.

$$g_{(1)} = Z \text{라고 할 때, } |g_{(1)}|m_1 = r_1 = a_1$$

$$g_{(2)} = (g_{(1)} - a_1)/m_1, |g_{(2)}|m_2 = a_2$$

$$g_{(j+1)} = (g_{(j)} - a_j)/m_j,$$

$$|g_{(j+1)}|m_{j+1} = a_{j+1} \quad (j=1, \dots, n-1, n: \text{정수}) \quad (12)$$

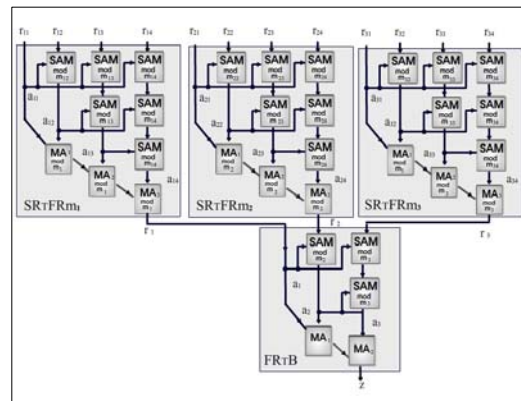


그림 3. MRC를 사용한 DRNS용 3D RTB변환기(n=3, kn=4)

식(12)는 전 단계 결과와 계수와의 감산 후 곱의 역에 의한 승산으로 이루어지므로, SAM (Subtraction and Multiplication) 연산 구조를 기본으로 사용하여 변환기를 설계할 수 있다. 모듈리가 3개(n=3)이고, 2차 잉여수의 모듈리가 4개(k₁=k₂=k₃=4)인 경우, MRC를 사용한 DRNS용 RTB변환기를 그림 3.에 나타내었으며, 3차원 어레이 프로세서(Array Processor)의 구조를 갖는다. 그림 3.에서 FR

TB는 식 (12)의 연산구조인 LUT(Look Up Table)를 사용한 SAM (Subtraction and Multiplication)을 기본으로 구성한다. 생성된 a_1 과 a_2 는 첫 번째 MA1(Multiplication and Addition)에서 $a_2m_1 + a_1$ 의 연산이 수행되고, 이 결과와 a_3 의 입력에 따라 $a_3m_1m_2 + MA1$ 연산 결과는 MA2에서 출력한다. SAM은 $m_i m_j \log_2 m_i$ 비트의 크기를 가지며, MA1은 $m_1 m_2 (\lceil \log_2(m_1 m_2 - 1) \rceil + 1)$ 비트의 크기이고 MA2는 $(M - m_3)(\lceil \log_2(M - 1) \rceil + 1)$ 비트의 크기를 갖는다.

DRNS에서 SRTFR $_{m_i}$ 의 표현은 III의 식(6)과 동일한 조건에서 식 (13)과 같이 표현할 수 있다.

$$|r_i|_{m_i} = \left\| a_{i, kn} \prod_{j=1}^{kn-1} m_{ij} + \dots + a_{i3} m_{i1} m_{i2} + a_{i2} m_{i1} + a_{i1} \right\|_{M_2} |m_i| \quad (13)$$

$a_{i, kn} \prod_{j=1}^{kn-1} m_{ij} + \dots + a_{i3} m_{i1} m_{i2} + a_{i2} m_{i1} + a_{i1} = A$ 라 하면, 식(13)은 식(14-1)과 같이 표현되고 식(14-2)의 조건에 따라 식(14-3)과 같이 표현될 수 있으므로, 식(15)와 같이 표현이 가능하며, SRTFR $_{m_i}$ 설계 시 1차 모듈리 m_i 에 따라 MA mod m_i 의 출력이 정해지므로 CRT를 사용한 MA보다 연산기의 크기를 감소시킬 수 있다.

$$|r_i|_{m_i} = |A|_{M_2} |m_i| \quad (14-1)$$

$$0 \leq A < M_2 - 1 \quad (14-2)$$

$$|A|_{M_2} |m_i| = |A|_{m_i} \quad (14-3)$$

$$|r_i|_{m_i} = \left\| a_{i, kn} \prod_{j=1}^{kn-1} m_{ij} |m_i| + \dots + |a_{i3} m_{i1} m_{i2}|_{m_i} + |a_{i2} m_{i1}|_{m_i} + |a_{i1}|_{m_i} \right\|_{m_i} \quad (15)$$

SRTFR $_{m_i}$ 의 경우, $k_1=k_2=\dots=kn$ 이라면 식

(12)와 같이 순환식으로 표시할 수 있으며,

$$g_{i(1)} = r_i \text{라고 할 때,}$$

$$|g_{i(1)}|_{m_{i1}} = r_{i1} = a_{i1}$$

$$g_{i(2)} = (g_{i(1)} - a_{i1}) // m_{i1}, |g_{i(2)}|_{m_{i2}} = a_{i2}$$

$$g_{i(k+1)} = (g_{i(k)} - a_{ik}) // m_{ik},$$

$$|g_{i(k+1)}|_{m_{i(k+1)}} = a_{i(k+1)} \quad (k=1, \dots, kn-1, kn: \text{정수}) \quad (16)$$

이 연산은 SAM mod m_{ik} 에서 연산이 수행되며, MA1 mod m_i 에서 $|a_{i2} m_1 + a_{i1}|_{m_i}$ 연산을 수행하며, MA $_{kn}$ mod m_i 에서 전단계의 출력

과 $|a_{i, kn} \prod_{j=1}^{kn-1} m_{ij}|_{m_i}$ 의 연산이 이루어진다. 그림

3에서 SAM mod m_{ik} 의 LUT는 $m_{ik} m_{ij} \log_2 m_{ik}$ 비트의 크기이며, MA1 mod m_i 의 LUT는 $m_{i1} m_{i2} \lceil \log_2 \{(m_{i1} m_{i2} - 1)\} + 1 \}$ 비트의 크기이며, MA2 mod m_i 의 LUT는 $(m_{i1} m_{i2}) m_{i3} \{ \lceil \log_2 (m_i - 1) \rceil + 1 \}$ 의 크기를 갖는다. MA3 mod m_i 의 LUT는 $m_i m_{i4} \{ \lceil \log_2 (m_i - 1) \rceil + 1 \}$ 의 크기를 갖는다.

V. 2D구조의 MRC 및 CRT를 사용한 DRNS용 RTB변환기 설계

IV장의 그림 3의 MRC를 사용한 DRNS용 RTB변환기는 3차원 구조를 갖는다. 본 논문에서는 연산기의 크기를 감소시키기 위하여 그림 4와 같이 SRTFR 연산부를 $[0,0,1]$ 방향으로 투영하여 2차원 구조의 변환기로 설계하고자 한다. 그림 3에서 SRTFR $_{m_1} \sim$ SRTFR $_{m_3}$ 에서 SAM mod $m_{1k} \sim$ SAM mod m_{3k} 는 동일한 구조를 가지므로 그림 4와 같이 투영하여 그림 5의 SRTFR $_{m_p}$ 의 SAM mod m_{ik} 를 설계할 수 있다. 그러나 계수의 승산과 중간 결과의

가산을 수행하는 MA는 1차 RNS의 모듈리 값에 따라 잉여수 연산을 수행 하므로,

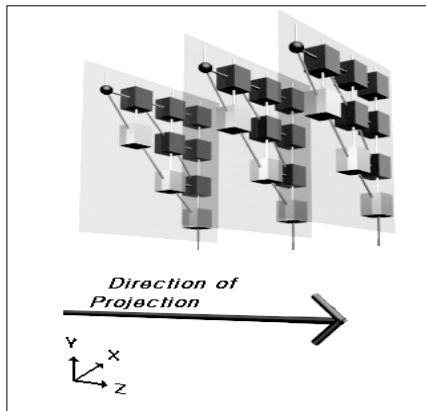


그림 4. 3D구조의 DRNS용 RTB변환기의 2D 투영

1차 RNS의 모듈리 별 부호를 p 라 할 때 i 번째 MA는 $MA_i \bmod m_p$ 의 연산을 수행하도록 설계한다. 그런데 $MA_i < m_p$ 인 경우는 출력은 MA_i 가 되어, 모듈리 별 연산기가 필요 없고 다음 MA_{i+1} 에 전달 값도 m_p 보다 작으므로 연산기 크기가 감소하게 된다. 설계된 RTB 변환기는 파이프라인 구조로 설계 되었으며, 첫 번째 계수 $a_{p1} = r_{p1}$ 이지만 데이터 전달의 동기를 맞추기 위하여 레지스터가 추가 되었다.

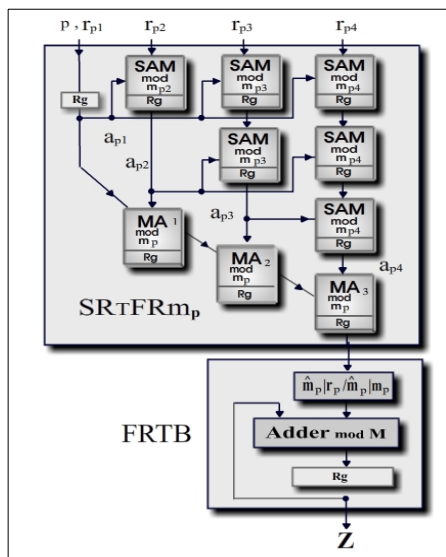


그림 5. 2D구조의 MRC 및 CRT를 사용한 DRNS용 RTB변환기

1차 잉여수를 2진수를 변환하는 FRTB는 IV. 장 그림3의 MRC를 사용하는 경우 CRT연산을 사용하는 경우와 비교하여 연산기의 크기가 증가하게 되므로, FRTB는 CRT 연산을 사용하여 SRTFR의 결과에 따라 $\widehat{m}_p | r_p // \widehat{m}_p | m_p$ 연산을 LUT를 사용하여 수행하고 모듈리 가산기 Adder modM에서 누적하도록 설계하여 RTB 변환기를 설계한다.

VI. 실험 및 고찰

본 논문에서 설계한 MRC를 사용한 DRNS용 RTB변환기에 대한 논리적 검증은 일반적인 HLL과 Xilinx FPGA용 시뮬레이터를 사용하였다. $n=3, k_1=k_2=k_3=4$ 이며, 1차 모듈리 $\{15, 16, 17\}$ 이고 2차 RNS 모듈리가 $\{3, 4, 5, 7\}$ 일 때, 본 논문에서 제안된 그림 5의 2D 구조의MRC 및 CRT를 사용한 DRNS용 RTB 변환기의 입출력신호를 그림 6.에 16진수로 나타내었다. PI1은 모듈리 구분 신호이고, RT01, RTT1, RTH2, RTF2는 2차 RNS의 표현이며 이 신호가 $\{0, 0, 0, 0\}, \{1, 1, 1, 1\}, \{2, 2, 2, 2\}$ 일 때 6 clock후 ZOUT11에서 변환된 출력 FE1₁₆을 생성한다. 다음 3clock 이후 2차 RNS의 표현 $\{0, 3, 3, 3\}, \{1, 0, 4, 4\}, \{2, 1, 0, 5\}$ 의 변환된 출력 FE4₁₆를 생성하며, XCLR신호에 따라 3clock 마다 누진 출력을 초기화 한다.

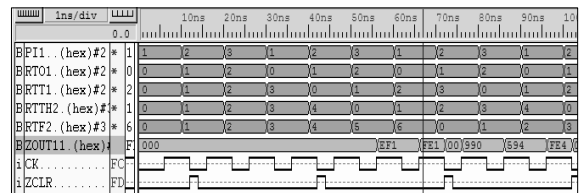


그림 6. 2D구조의 MRC 및 CRT를 사용한 DRNS용 RTB변환기의 입출력 신호

표 1. DRNS용 SRTFR변환기 크기 비교

종류	SRTFR변환기 크기(bits)
CRT	$\sum_{i=1}^p (\prod_{j=1}^k m_{ij}) (\lceil \log_2 m_i - 1 \rceil + 1)$
3D MRC 그림 3.	$\sum_{i=1}^p [\sum_{d=2}^k \sum_{j=1}^{d-1} m_{id} m_{ij} (\lceil \log_2 (m_{id} - 1) \rceil + 1) + m_{i1} m_{i2} \lceil \log_2 \{ (m_{i1} m_{i2} - 1) \rceil + 1 \} + (m_{i1} m_{i2}) m_{i3} \{ \lceil \log_2 (m_i - 1) \rceil + 1 \} + \sum_{j=4}^k m_i m_{ij} \{ \lceil \log_2 (m_i - 1) \rceil + 1 \}]$ (단, $k \geq 4, p \geq 3, m_{ik} < m_i$)
2D MRC 그림 5.	$\sum_{d=2}^k \sum_{j=1}^{d-1} m_{pd} m_{pj} (\lceil \log_2 (m_{pd} - 1) \rceil + 1) + m_{p1} m_{p2} \lceil \log_2 \{ (m_{p1} m_{p2} - 1) \rceil + 1 \} + \sum_{i=1}^p [(m_{i1} m_{i2}) m_{i2} \{ \lceil \log_2 (m_i - 1) \rceil + 1 \} + \sum_{j=4}^k m_i m_{ij} \{ \lceil \log_2 (m_i - 1) \rceil + 1 \}]$ (단, $k \geq 4, p \geq 3, m_{ik} < m_i$)

종류	SRTFR변환기 크기(bits)	
	1차 RNS 모듈리 {15, 16, 17} 2차 RNS 모듈리 {3, 4, 5, 7}	1차 RNS 모듈리 {29, 31, 32} 2차 RNS 모듈리 {4, 5, 7, 9}
CRT	5880	18900
3D MRC 그림 3.	3530	9632
2D MRC 그림 5.	2672	7165

표 2. 모듈리에 따른 DRNS용 SRTFR변환기 크기 비교

1차 RNS 모듈리의 수를 p, 2차 RNS 모듈리의 수를 k라 하고, 2진으로 변환될 데이터가 n개 일 때, 1개 LUT의 연산 시간을 clock주기로 사용하는 경우 clock 신호발생 시간을 t라 하면, CRT만을 사용하는 경우는 2t clock이지만 연산기의 크기가 증가하게 되는 문제점을 갖는다. 3D구조의 MRC만을 사용하는 경우는 (k+p)t clock이고 그림 3을 파이프라인으로 설계하는 경우는 (k+p)+t clock까지 속도를 향상시킬 수 있지만, 그림 5보다 연산기가 커지는 문제점을 갖는다. 2D구조의 MRC와 CRT를 사용한 연산시간은 (k+n-1)+nt clock

이어서 지연시간을 갖지만 작은 크기의 연산기로 처리가 가능하다. LUT를 사용하여 설계하였을 때 SRTFR변환기의 비교를 표 1.에 나타내었으며, 모듈리에 따른 비교표는 표 2.에 나타내었다. 그림 5의 변환기는 CRT를 사용한 경우보다 1/2.3~1/2.6 만큼 연산기가 감소하였으며, 그림 3의 변환기와 비교하여 1/1.3만큼 변환기의 크기가 감소되었음을 알 수 있다.

Ⅶ. 결론

영상처리, 신경망, 디지털 신호처리 분야등과 같이 고속의 전용 연산이 요구되는 분야에서 잉여수계를 사용한 연산기는 연산처리 속도를 향상시키는 중요한 방법 중의 하나이다. DRNS는 이러한 잉여수 연산기의 연산기 크기를 감소시키고, 모듈리를 소수로 만 구성하는 경우에도 연산기의 크기를 감소시킬 수 있는 장점을 가지고 있지만 비가중치수체계의 잉여수를 2진수로 변환 시 2번의 변환과정이 필요하므로 연산기의 크기를 증가시키는 문제점을 갖는다.

DRNS에서 1차 RNS의 잉여수를 이진수(FRTB)로 변환하는 방법은 많은 연구가 있었다. CRT를 사용한 고속화 방법은 모듈리에 제한성을 가지므로, 2차 잉여수를 1차 잉여수(SRTFR)로의 효율적인 변환에 적합하지 않다. 그러므로 본 논문에서는 2차 잉여수를 1차 잉여수로 변환하는 SRTFR 변환기의 연산기 크기를 감소시키기 위하여 MRC를 사용하여 설계하였으며, FRTB는 CRT를 사용하여 설계하였다. MRC를 사용한 SRTFR 변환기는 각 항 별로 1차 잉여수의 모듈리를 적용할 수 있으므로 SRTFR변환기의 크기를 감소시킬 수 있었다. 잉여수계를 사용한 시스템 규모에 따라 연산속도가 중요시 되는 경우에는 그림 3.의 3D구

조의 SRTFR 변환기로 사용할 수 있으며, 그림 5.에 설계된 변환기는 파이프라인 구조를 갖도록 2차원 구조로 설계하였으며, 지연시간은 있으나 그림 3.의 3차원 구조의 SRTFR 변환기와 비교하여 적은 크기의 연산기로 변환기가 설계되었으므로 연산기 크기를 최소화하는 경우에 유용하게 사용할 수 있다.

앞으로 SRTFR의 연산기 크기 및 속도를 보다 효율적으로 향상시키기 위하여 MA의 구조에 대한 연구가 계속되어야 할 것이다.

참고문헌

- [1] K.D.Weinmann, M.A.Soderstrand and S.Shebani, "Evaluation of New Hardware for a High-speed, Digital, Adaptive Filter Using the Residue Number System", *16th Asilomar Conference on Circuits, Systems, and Computers*, pp. 187-191, 1982.
- [2] M.A. Soderstrand, C. Vernia and Jui-Hua Chang, "An Improved Residue Number System Digital to Analog Converter", *IEEE Trans. on Circuits and Systems*, Vol. CAS-30, pp. 903-909, December, 1983.
- [3] Michael A. Soderstrand, Bhaskar Sinha, "A Pipelined Recursive Residue Number System Digital Filter", *IEEE Transactions on Circuits and Systems*, Vol. CAS-31, No. 4, April, 1984.
- [4] 윤현식, 조원경, "잉여수계를 이용한 디지털 신경망회로의 실현", *전자공학회논문지*, 제 30권, B편, 제 2호, pp. 44-50, 2월, 1993.
- [5] 김용성 외 1, "고속 그래픽 처리를 위한 잉여수계 승산기 설계에 관한 연구", *전자공학회논문지*, 제 33권, B편, 제 1호, pp. 25-37, 1월, 1996.
- [6] Nicholas S. Szab, Richard I. Tanakas, *Residue Arithmetic and its Applications to Computer Technology*, McGraw-Hill, 1967.
- [7] 김용성, "MAC연산용 다중 잉여수계에 대한 연구", *여주대학 산업기술 연구소 논문집*, 제1권, pp.317-332, 11월, 1995.
- [8] 김용성 외 1, "고속 그래픽 처리를 위한 잉여수계 승산기 설계에 관한 연구" *대한전자공학회논문지*, 제 33권, B편, 제 1호, pp.25-37, 1월, 1996.
- [9] Stanishaw J. Piestrak, "A High-Speed Realization of Residue to Binary Number System Converter", *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, Vol. 42, No 10, pp. 661-662, October, 1995.
- [10] Khalid M. Ibrahim and Salam N. Salqum, "A Efficient Residue to Binary Converter Design", *IEEE Trans. on Circuits and Systems*, Vol. 35, No 9, pp. 1156-1158, September, 1998.

A Study on the design of First Residue to Second Residue Converter for Double Residue Number System

Young Sung Kim

Abstract

Residue Number System is used for the purpose of increasing the speed of processing in the many application parts of Image Processing, Computer Graphic, Neural Computing, Digital Signal Processing etc, since it has the characteristic of parallelism and no carry propagation at each moduli. DRNS has the twice RNS Conversion, it is used to decrease the size of the operator in RNS. But it has a weak point on the Second Residue to First Residue Conversion time.

So, in this paper SRTFR(Second Residue to First Residue) Converter using MRC(Mixed Radix Conversion) is designed to decrease the size of RTB(Residue to Binary) Converter. Since the proposed SRTFR Converter using MRC(Mixed Radix Conversion) has a pipeline processing. Also, modular operation is applied to at each partitioned SAM(Subtraction and Addition) and MA(Multiplication and addition). In the following study, the more effective design on MA is needed.

key words: RNS, DRS, MRC, modular operation