

論文94-31B-5-1

## 1차원 시스톨릭 어레이 프로세서를 이용한 고속 곡선 발생기에 관한 연구

### (A study on the High Speed Curve Generator Using 1-Dimensional Systolic Array Processor)

金龍成\*, 趙源敬\*

( Yong Sung Kim and Won Kyung Cho )

#### 要約

컴퓨터 그래픽에 있어서 도형은 선과 곡선으로 이루어진다. 고속의 곡선 발생기는 컴퓨터를 이용한 설계 및 시뮬레이션 등과 같이 실시간 그래픽 처리가 요구되는 분야에서 필수적인 요소이다. 그래픽 처리를 위한 기본형태는 행렬 연산으로 표현되므로, 본 논문에서는 베지어(Bezier)곡선 발생식의 순환성 및 행렬 표현식을 이용하여 고속의 베지어 곡선 발생기를 두가지 형태로 설계하며, B-스플라인(B-spline) 곡선 발생식의 배합함수(blending function)에 대한 상호 의존성을 사용하여 B-스플라인 곡선 발생기를 설계한다. 설계된 곡선 발생기와 참고문헌 [5], [6] 과의 성능비교 결과, 베지어 곡선의 행렬 표현식을 이용한 행렬·벡터 연산용 1차원 시스톨릭 어레이 프로세서(Systolic Array Processor)의 구조를 갖는 곡선 발생기가 효율적임을 알 수 있었다.

#### Abstract

In computer graphics, since objects are constructed by lines and curves, the high-speed curve generator is indispensable for computer aided design and simulation. Since the functions of graphic generation can be represented as a series of matrix operations, in this paper, two kind of the high-speed Bezier curve generator that uses matrix equation and a recursive relation for Bezier polynomials are designed. And B-spline curve generator is designed using interdependence of B-spline blending functions. As the result of the comparison of designed curve generator and reference [5], [6] in the operation time and number of operators, the curve generator with 1-dimensional systolic array processor for matrix vector operation that uses matrix equation for Bezier curve is more effective.

#### 1. 서론

1960년대에 시작된 컴퓨터 그래픽은 1980년대 이후 VLSI설계 기술이 발달함에 따라 급격히 발달하여 왔으며,

컴퓨터와 사용자간의 인터페이스 문제등과 같이 컴퓨터를 사용한 입출력 부분 및 설계 분야에 다양하게 사용되고 있다. 컴퓨터 그래픽에 있어서 도형은 점, 직선, 원 및 곡선으로 이루어지며, 직선 및 원은 곡선의 특수한 형태라할 수 있다. 그러므로 곡선발생은 컴퓨터를 이용한 설계, 시뮬레이션, 애니메이션, 외곽선 글자 생성등의 분야에 필수적이며, 이와 같은 분야를 실시간으로 처리하기 위하여 고속의 곡선발생기가 요구된다. [1], [2], [3]

\* 正會員, 慶熙大學校 電子工學科  
(Dept. of Elec. Eng., Kyunghee Univ.)  
接受日字 : 1994年 7月 10日

곡선을 발생하는 방법은 매개변수 식을 사용하는 곡선 근접화(curve fitting)방법과 비매개변수 식을 사용하는 방법으로 나누어진다. 비매개변수 식을 사용하는 곡선 발생식은 Lagrange, Spline 다항식이 있으며 [2] - [3], 주어진 조정점(control points)들 사이에서 진동하는 형태로 표현되는 단점을 갖는다. 그러므로 주어진 조정점의 연결선에 근접하는 곡선을 발생시키는 곡선 근접화 방법을 주로 사용한다. 곡선 근접화는 배합함수(blending function)를 사용하며, 베지어(Bezier)곡선 및 B-스프라인(B-spline)곡선이 대표적이다. [1], [3], [4] B-스프라인 곡선을 이용한 곡선 발생기 설계시 B-스프라인 곡선 발생식의 간략화 없이 적용하는 경우, 하드웨어의 크기가 증가하는 단점을 갖는다. [5], [6] 또한, 베지어 곡선의 식을 간략화하여 곡선 발생기를 설계하는 경우, 그래픽 프로세서에 적용시 유용성을 고려 하여야 한다. [6], [7]

그러므로, 본 논문에서는 고속의 베지어 곡선 및 B-스프라인 곡선 발생기를 설계하기 위하여 베지어 곡선 발생식의 순환성 및 B-스프라인 곡선의 배합함수에 대한 의존성을 고찰하고 [1], [4] 배합함수의 간략화를 사용하여 곡선 발생기를 설계한다. 또한 그래픽 처리를 위한 기본형태는 행렬식으로 표현할 수 있으므로 [1], [3], [4], [11], 행렬 연산용 프로세서를 주 연산기로 사용한 그래픽 프로세서에 적용할 수 있도록 하기 위하여 [8], [9], [10] 1차원 행렬·벡터( [4×4] · [4] ) 연산용 시스톨릭 어레이 프로세서를 사용한 베지어 곡선 발생기를 설계하고자 한다.

II. 곡선 발생의 기본 알고리즘

곡선 근사화에 사용되는 베지어 및 B-스프라인 곡선은 조정점들의 연결선에 접근된 곡선을 발생하는 방법으로 배합함수에 의해 곡선의 근접화 및 연속성이 이루어지며, 매개변수에 의한 식으로 표현된다.

1. 베지어 곡선

베지어 곡선의 배합함수는 매개변수 u와 조합(combination)계산으로 이루어지며, 조정점  $P_i(i=0, \dots, n)$ 와 매개변수  $u(0 \leq u \leq 1)$ 에 따라 발생된 곡선의 한점을  $P(u)$ 라 할 때 곡선 발생식은 식(1)과 같으며, 식(1)에 의해 발생된 베지어 곡선을 그림 1에 나타내었다.

$$P(u) = [x(u), y(u)], \quad P(u) = \sum_{i=0}^n P_i B_i, n(u) \quad (1)$$

$B_{i,n}(u) = c(n, i)u^i(1-u)^{n-i}$  이므로, 이차원 곡선 출력은

$$X(u) = \sum_{i=0}^n X_i B_i, n(u), \quad Y(u) = \sum_{i=0}^n Y_i B_i, n(u) \text{ 이 된다.}$$

(단,  $0 \leq u \leq 1$ , n : 조정점 수,  $P_i$ : 조정점 )

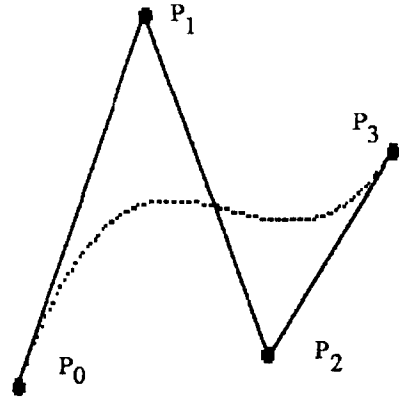


그림 1. 베지어 곡선 (조정점=4)  
Fig. 1. Bezier curve. (control points=4)

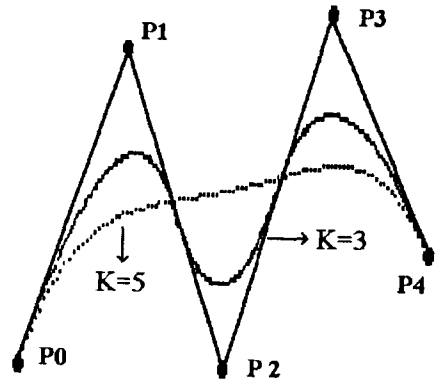


그림 2. 비스라인 곡선 (조정점=5, k=3, k=5)  
Fig. 2. B-spline curve. (control points=5 k= 3, k = 5 )

2. B-스프라인 곡선

스프라인 곡선은 1개의 조정점의 위치가 변동되면 전체 곡선이 변화되는 문제점을 갖으므로 이러한 단점을 배합함수  $N_{i,k}$ 를 사용하여 보완한 방법이 B-스프라인 곡선이다. 조정점  $P_i$ 와 매개변수 t에 따라 발생된 곡선의 한점을  $P(t)$ 라 할 때 곡선 발생식은 식(2)와 같다.

$$P(t) = \sum_{i=0}^n P_i N_i, k(u) \quad (0 \leq t \leq n - k + 2) \quad (2)$$

$P_i(0 \leq i \leq n)$ : 조정점, k: 매듭(knot)

또한 배합함수  $N_{i,k}(u)$ 는 다음과 같다.

$$N_{i,k}(t) = \begin{cases} 1 & : t_i \leq t \leq t_{i+1} \\ 0 & : \text{else} \end{cases} \quad (3-1)$$

$$N_{i,k}(u) = \frac{(t-t_i)N_{i,k-1}(t)}{t_{i+k-1}-t_i} + \frac{(t_{i+k}-t)N_{i+1,k-1}(t)}{t_{i+k}-t_{i+1}} \quad (\text{단, } 0/0 = 0) \quad (3-2)$$

$$\text{또한 } t_i = \begin{cases} 0 & \text{if } i < k \\ j-k+1 & \text{if } k \leq i \leq n \quad (0 \leq i \leq n+k) \\ n-k+2 & \text{if } i > n \end{cases} \quad (3-3)$$

B-스프라인 곡선은 매듭(knot) k에 따라 곡선의 굴곡도가 변화되며, k=3.5인 경우의 B-스프라인 곡선을 그림 2에 나타내었다.

### III. 베지어 곡선 발생식의 순환성을 사용한 곡선 발생기의 설계

식(1)의 베지어 곡선발생식은 조합계산 및 매개변수의 i제곱의 연산으로 표현되므로 이식을 사용하여 곡선 발생기를 설계하는 경우 연산부의 크기가 증가되고 연산속도가 저하된다. 그러므로 식(1)을 식(4)와 같이 순환성을 갖도록 표현하고<sup>[4]</sup>, 식 (4)를 이용하여 베지어곡선 발생기를 설계한다. 식 (1)의 초항과 말항을 분리하면

$$P(u) = (1-u)^n P_0 + \sum_{i=1}^{n-1} c(n,i)u^i(1-u)^{n-i} + u^n p_n \text{ 이 되며,}$$

$$P(u) = (1-u)^n P_0 + \sum_{i=1}^{n-1} c(n-1,i)u^i(1-u)^{n-i} P_i + \sum_{i=1}^{n-1} c(n-1,i-1)u^{i-1}(1-u)^{n-i} P_i + u^n p_n \text{ 이며,}$$

$$(\because c(n-i) = c(n-1,i) + c(n-1,i-1))$$

$$P(u) = (1-u)^n [(1-u)^{n-1} P_0 + \sum_{i=1}^{n-1} c(n-1,i)u^i(1-u)^{n-i} P_i + u \sum_{i=1}^{n-1} c(n-1,i-1)u^{i-1}(1-u)^{n-i} P_i + u^{n-1} p_n]$$

$$P_{0,n} = P_{0,n-1}(u) + u [P_{1,n}(u) - P_{0,n-1}(u)] \quad (P_{0,n}(u) \text{는 } P_0 \sim P_n \text{으로 이루어진 다항식}) \quad (4)$$

로 표현된다. 그러므로 본 논문에서는 식(4)의 순환성을 이용하여 베지어곡선기를 설계하고자 한다.

1. 조정점이 4개인 경우 베지어 곡선 발생기의 설계  
조정점이 4개인 경우 식(4)의 순환성을 이용하면 다음과 같이 곡선 발생점  $P_{03}$ 를 구할 수 있다. 4개의

조정점을  $P_0, P_1, P_2, P_3$ 라 할 때 (단,  $P_{00}(u)=P_0, P_{11}(u)=P_1, P_{22}(u)=P_2, P_{33}(u)=P_3$ )

$$\begin{aligned} P_{01}(u) &= P_{00}(u) + u [P_{11}(u) - P_{00}(u)] \\ P_{12}(u) &= P_{11}(u) + u [P_{22}(u) - P_{11}(u)] \\ P_{23}(u) &= P_{22}(u) + u [P_{33}(u) - P_{22}(u)] \text{ 이며,} \\ P_{02}(u) &= P_{01}(u) + u [P_{12}(u) - P_{01}(u)] \\ P_{13}(u) &= P_{12}(u) + u [P_{23}(u) - P_{12}(u)] \text{ 이므로} \\ P_{03}(u) &= P_{02}(u) + u [P_{23}(u) - P_{02}(u)] \text{ 이다.} \end{aligned}$$

즉 4개의 조정점에 의해  $P_{01}, P_{12}, P_{23}$  이 값에 의하여  $P_{02}, P_{13}$ 를 구할 수 있으며,  $P_{02}, P_{13}$ 에 의해 1개의 곡선 발생점  $P_{03}$ 를 구할 수 있다. 이와 같은 관계에 의하여 2차원 어레이 구조를 갖는 곡선 발생기를 그림 3 (a)에 나타내었다. 여기서  $u$ 는  $0 \leq u \leq 1$ 이므로 곡선 구성점의 수가 m개일 때  $u$ 는  $u_k = k/m$  ( $k = 0, \dots, m$ )가 된다. 설계된 곡선 발생기는 2 스텝후 첫번째 출력이 발생된다. 설계된 병렬 곡선 발생기의 하드웨어 복잡도 및 크기를 감소시키기 위하여 1차원 곡선 발생기를 행방향으로 사상(mapping)하여 그림 3(b)와 같이 1차원 어레이로 설계한다. 그림 3 (b)의 1차원 곡선 발생기는 첫번째 기본 처리요소(PE: processing element)에서  $P_{01}, P_{12}, P_{23}$ 를 구하며, 두번째 기본 처리요소에서  $P_{02}, P_{13}$ 를 구하고, 다시 이 값에 따라 3번째 기본 처리요소에서  $P_{03}$ 를 구하므로 5스텝후 첫번째 출력이 생성되고 첫번째 출력후 3스텝마다 출력이 생성된다.

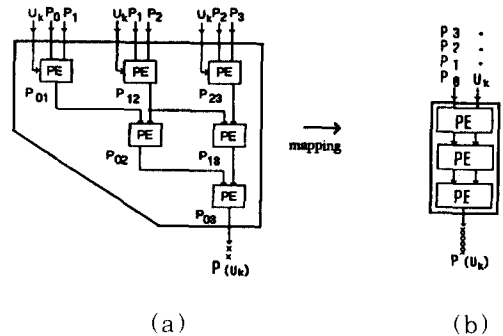


그림 3. 베지어곡선 발생기(조정점=4)  
(a) 2차원 어레이 (b) 1 차원 어레이

Fig. 3. Bezier curve generator(control points=4)  
(a) 2-dimensional Array  
(b) 1-dimensional Array.

2. 조정점이 3개인 경우 베지어 곡선 발생기의 설계  
조정점이 3개인 경우 곡선발생기는 조정점이 4개인

경우와 동일한 방식으로 설계할 수 있으며, 3개의 조정점을  $P_0, P_1, P_2$ 라할 때 곡선 발생점은  $P_{02}$ 이며, 그림 4 (a)와 같이 2차원 어레이로 설계할 수 있다.

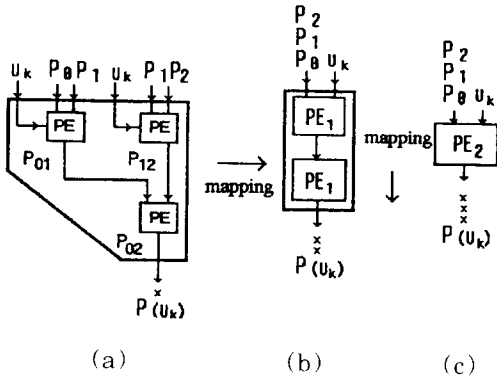


그림 4. 베지어곡선 발생기(조정점=3)  
 (a) 2차원 어레이 발생기 (b) 1 차원 어레이 발생기 (c) 두번 사상된 기본 처리요소

Fig 4. Bezier curve generator(control points=3)  
 (a) 2-dimensional Array (b) 1-dimensional Array (c) double mapped PE.

2 스텝에 첫번째 출력이 발생되고 3 스텝 부터는 연속적으로 출력이 발생한다. 그림 4 (b)의 1차원 발생기는 그림 4(a)를 행방향으로 사상한 것으로 2스텝마다 새로운 출력이 생성된다. 그림 4 (b)를 열방향으로 사상하면 그림 4 (c)와 같이 1개의 기본 처리요소(PE2)로 설계할 수있으며, 4 스텝마다 새로운 출력을 생성할 수 있다. 그러므로 그림 4 (c)의 기본 처리요소(PE2) 4개를 사용하여 그림 5 (a)와 같이 구성할 수 있다.

3스텝후 첫번째 기본 처리요소에서 출력이 발생되고, 4 스텝에서는 두번째 기본 처리요소에서 출력이 발생하므로 매개변수  $u_k$ 에 대한 출력은  $k \bmod 4$  번째 기본 처리요소에서 출력이 발생할 수 있다. 그림 5 (a)의 경우 매개변수  $u_k$ 는 행방향으로 입력되며, 이 때 입력된  $P_i$ 와의 곱에 의하여 곡선 발생점을 구하고, 연산된 결과는 각 기본 처리요소마다 출력된다. 그림 5 (b)는 곡선 생성점이 열방향으로 출력되도록 기본 처리요소(PE3)를 사용하여 설계한 곡선 발생기이다.

IV. B-스프라인 곡선발생기의 설계

B-스프라인 곡선은 배합함수를 사용하여 곡선을 발생한다. 식(3)에 의해서 배합함수를 구하는 경우 여러단계의 연산이 요구되므로, 본 논문에서는 매듭  $k=3$ 인 경우 배합함수의 종속성<sup>(1), (4)</sup> 및 간략화 과정을 이용하여 곡선 발생기를 설계하고자 한다.

$k=3$ 인 경우 배합함수  $N_{i,k}(u)$ 는  $N_{i+1,k-1}(u)$ 와  $N_{i-1,k-1}$ 에 의해서 구해지므로 그림 6(a)와 같은 종속관계로 표현된다. 또한 구간이  $t_j \leq t < t_{j+1}$ 인 경우 식 (3-1)에 의해서  $k=1$ 일 때  $N_{i,1}$ 만 1이므로  $N_{i,1}, N_{i,2}, N_{i-1,2}, N_{i,3}, N_{i-1,3}, N_{i-2,3}$ 에 대한 연산부만 필요하고, 그림 6 (b)와 같이 종속관계가 표시되므로 조정점  $n=5$  (조정점 6개), 매듭  $k=3$ 인 경우 식(5)와 같이 표시된다.

$$P(t) = P_{i-2}N_{i-2,3}(t) + P_{i-1}N_{i-1,3}(t) + P_iN_{i,3}(t) \quad (n=5, i=2, \dots, n)$$

또는,  $P_i(t) = P_{i-1}(t) + P_{i-1}N_{i-1,3}(t)$

$$(i=0, \dots, 2, n=5, i=2, \dots, n, P_0(u)=0) \quad (5)$$

식(5)에 의해 구성된 B-스프라인 곡선 발생기의 구조를 그림 6 (c)에 나타내었으며, 각 기본 처리요소에서는 승산과 가산을 수행하는 구조로 구성한다.

본 논문에서는 그림 6(c)에 곡선 발생기의 연산부의 크기를 감소시키기 위하여 배합함수  $N_{i,k}$ 를 다음과 같이 간략화 한다.  $k=3, n=4$ 인 경우,  $t$ 의 구간은

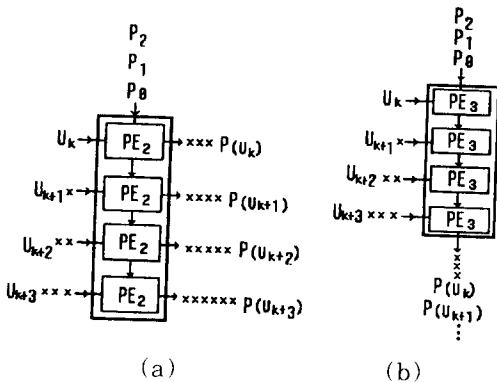


그림 5. 연속성을 갖는 1차원 베지어 곡선 발생기 (조정점=3)(a) 출력이 행 방향인 경우 (b) 출력이 열 방향인 경우

Fig. 5. 1-dimensional bezier curve generator (control points=3)  
 (a) in case of output for row direction (b) in case of output for column direction.

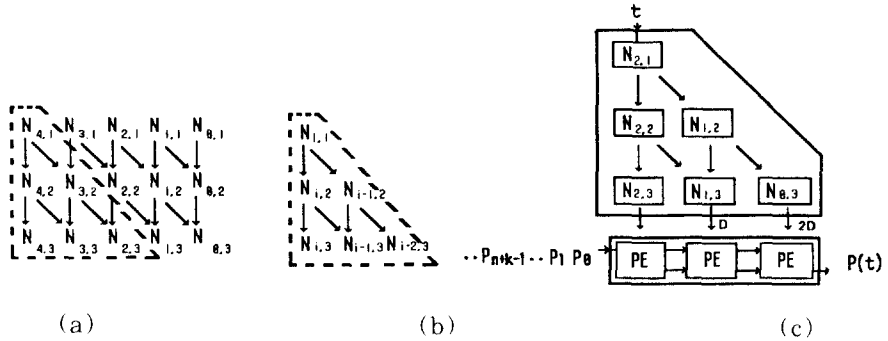


그림 6. B-스프라인 곡선 근사화시 배합함수(Ni,k)의 종속관계  
 (a) k=3, n=4인 경우 (b) n=i, k=3인 경우 (c) B-스프라인 곡선 발생기  
 Fig. 6. The dependency of blending function(Ni,k) in B-spline curve generation  
 (a) In case of k=4, n=4. (b) In case of k=4, n=i. (c) B-spline curve generator.

$t_0 \sim t_7$ 이며, 식(3-2)에 의해서

$$t_0 = t_1 = t_2 = 0, t_3 = 1, t_4 = 2, t_5 = 3, t_6 = t_7 = 4 \quad (6)$$

이므로 실제로 사용되는 구간은  $t_i \leq t \leq t_{i+1}$  ( $2 \leq i \leq 5$ )이며, 곡선 구성점의 수가 m개일 때 곡선 매개변수 t를  $ts=4s/m$  ( $s = 0, \dots, m$ )로 표현할 수 있으며 t를 정수부와 소수부로 나누어 표시하면

$$t = I + u = t_i + u \quad (I : 0 \leq I \leq 4 \text{의 정수}, u : 0 \leq u \leq 1 \text{의 소수}, 2 \leq i \leq 5) \quad (7)$$

이 되며, 식(6)과 식(7)을 사용하면 k=3인 경우  $N_{i,k}$ 를 다음과 같이 나타낼 수 있다.

$$\begin{aligned} N_{i,1}(t) &= 1 \quad (t_i \leq t \leq t_{i+1}) \\ N_{i,2} &= 0 + \frac{t-t_i}{t_{i+1}-t_i} N_{i,1}(t) = u, \\ N_{i-1,2} &= \frac{t_{i+1}-t}{t_{i+1}-t_i} N_{i,1}(t) + 0 = 1-u \\ N_{i,3} &= \frac{t-t_i}{t_{i+2}-t_i} N_{i,2}(u) + 0 = u^2 / s_1 \\ &\quad (\text{if } i=5 \text{ then } s_1 = 1 \text{ else } s_1 = 2) \end{aligned} \quad (8-1)$$

$$\begin{aligned} N_{i-2,3} &= 0 + \frac{t_{i+1}-t}{t_{i+1}-t_{i-1}} N_{i-1,2}(u) = (1-u)^2 / s_2 \\ &\quad (\text{if } i=2 \text{ then } s_2 = 1 \text{ else } s_2 = 2) \end{aligned} \quad (8-2)$$

$$N_{i-1,3} = 1 - N_{i,3} - N_{i-2,3} \left( \because \sum_{j=0}^{k-1} N_{i-j,k} = 0 \right) \quad (8-3)$$

식(5)와 식(8)의 결과에 따라 설계한 B-스프라인 곡

선 발생기를 그림 7에 나타내었다.

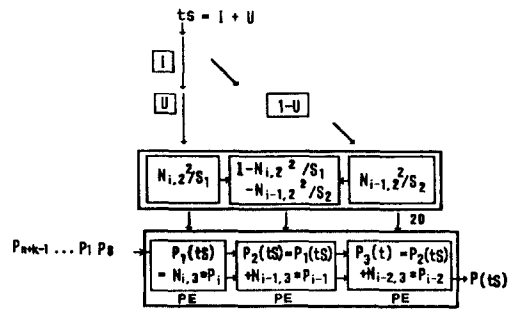


그림 7. B-스프라인 곡선 발생기  
 Fig. 7. B-spline curve generator.

그림 7에서 매개변수 t는 정수 I와 소수 t로 분리하여 입력하고 식(8-1)과 식(8-2)에 의해  $s_1$ 과  $s_2$ 를 결정한다. 또한 식(8-3)에 의해서  $N_{i,k}(u)$  연산부는 두개의 승산기와 감산기 만으로 구성할 수 있으며, 식(5)에 의해 연산된  $N_{i,k}(u)$ 와  $P_i$ 와의 승산 및 가산은 1차원 시스템릭 어레이 구조로 설계한다. 2차원 도면에서 곡선을 발생하는 경우  $P_i$ 는  $x_i, y_i$ 로 구분되므로 설계된 곡선 발생기에서 두배의 기본 처리요소 가 소요된다.

#### IV. 벡터 행렬 및 벡터 벡터 시스템릭 어레이 프로세서를 이용한 베지어 곡선 발생기의 설계

그래픽 프로세서 설계시 그래픽 기본요소를 생성하기 위한 각각의 프로세서를 설계하는 것보다 하나의

주 연산기를 통하여 그래픽 기능을 수행하는 것이 바람직하다. 행렬 연산기를 주 연산기로 사용하는 그래픽 프로세서에 적용할 수 있도록 하기 위하여 본 논문에서는 1차원 행렬·벡터 연산용 시스톨릭 어레이 프로세서를 주 연산기로 사용하여 베지어 곡선 발생기를 설계하고자 한다. 그림 8에 1차원 행렬·벡터 연산용 시스톨릭 어레이 프로세서를 사용한 B-스프라인 곡선 발생기의 기본구조를 나타내었으며, 행렬·벡터 ( $[4 \times 4] \cdot [4]$ ) 시스톨릭 어레이와 벡터·벡터 ( $[4] \cdot [4]$ ) 어레이로 구성된다. 조정점이 4개인 경우 베지어곡선의 행렬 표현식은 식(9)와 같다.

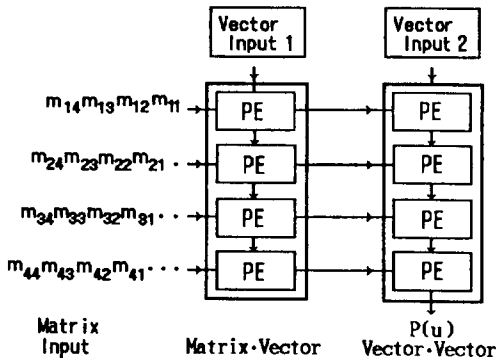


그림 8. 행렬·벡터 시스톨릭 어레이를 이용한 곡선 발생기의 기본형태

Fig. 8. Fundamental model of curve generator using Matrix Vector Systolic Array Processor.

$$G = [P_0 \ P_1 \ P_2 \ P_3]$$

$$M = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \quad U = \begin{bmatrix} u^3 \\ u^2 \\ u^1 \\ u^0 \end{bmatrix}$$

$$GM = G \cdot M = [gm_1, gm_2, gm_3, gm_4]$$

$$P(u) = G \cdot M \cdot U \quad P_k(u) = P_{k-1}(u) + GM_k \cdot U_k$$

$$(k = 1, \dots, 4, P_0(u) = 0, 0 \leq u \leq 1) \quad (9)$$

식(9)를 그림 8의 곡선 발생기에 적용하는 경우, 행렬입력부에 행렬 M가 입력되며, 벡터 입력부1에 벡터 G를 입력하고 벡터입력 2에 벡터 U를 입력한다. 이 때 곡선 구성점은 행렬·벡터 연산결과와 열 벡터와의 연산에 의해서 생성된다.

식 (9)의 G·M 연산은 1회만 수행되므로, G·M 연산을 행렬·벡터 연산부에서 1회 수행한 결과와 U 벡터와의 연산을 통하여 8 스텝후 첫번째 출력이 발

생하며, 이후 연속적으로 곡선의 구성점을 발생할 수 있다. 그러나 조정점  $P_i$ 는  $x_i, y_i$ 의 이차원 성분의 벡터이므로 2차원 평면에서 곡선을 발생 시키기 위하여 2개의 곡선 발생기가 요구된다.

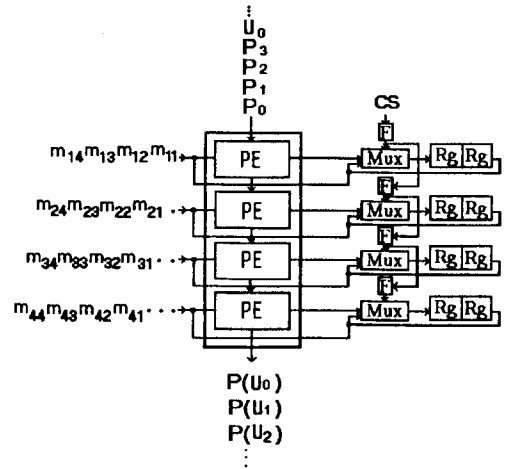


그림 9. 행렬·벡터 및 벡터·벡터 시스톨릭 어레이를 이용한 베지어 곡선 발생기

Fig. 9. Bezier curve generator using Matrix Vector, Vector Vector Systolic Array Processor.

그러므로 본 논문에서는 그림 9와 같이  $G \cdot M$  연산 시  $x_i, y_i$  성분에 따른 연산결과  $G_x \cdot M_x$ 와  $G_y \cdot M_y$ 가 생성될 때까지는 제어신호  $cs=1$ 에 의해 출력결과를 레지스터에 저장하고, 제어신호  $cs=0$ 에 의해 레지스터에 저장된  $G_x \cdot M_x$ 와  $G_y \cdot M_y$ 를 회전하여 행렬 데이터 입력부에 입력하여 벡터 U와의 연산을 수행함으로써 연속적인 출력을 얻을 수 있다. 이때 곡선 발생기의 기본 처리요소는 행렬·벡터 연산 및 벡터·벡터 연산이 가능한 구조로 설계한다.

## VI. 곡선 발생기의 기본 처리요소 설계

설계된 베지어 및 B-스프라인 곡선 발생기의 기본 처리요소를 다음과 같이 설계한다.

### 1. 베지어 곡선 발생기의 기본 처리요소 설계

그림 3(a)와 그림 4(a)의 베지어 곡선 발생기의 기본 처리요소는 식 (4)에 따라 입력  $P_{i,j}, P_{i+1,j+1}$ 의 차와 u의 승산으로 표시되며, 그림 10(a)와 같이 구성한다. 그림 3(b)와 그림 4(b)의 베지어 곡선 발생기

의 기본 처리요소는 식 (4)에 따라 입력  $P_{i,j}$ 와  $P_{i+1,j+1}$ 가 순서적으로 입력되므로 입력단에 두개의 레지스터를 사용하며, 이 두값의 차와  $u$ 와의 승산을 수행하므로써

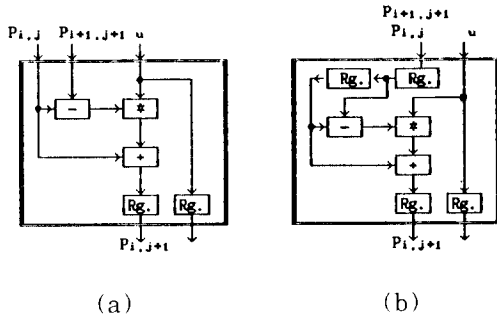


그림 10. 베지어 곡선 발생기의 기본 처리요소. (조정점=4) (a) 2차원 발생기의 기본 처리요소 (b) 1차원 발생기의 기본 처리요소

Fig. 10. PE of bezier curve generator

(control points=4)

(a) PE of 2-dimensional generator

(b) PE of 1-dimensional generator

그림 10(b)와 같이 기본 처리요소를 구성한다. 조정점이 3개인 경우, 그림 4(c)에 설계된 두번 사상한 기본 처리요소를 사용한 1차원 베지어 곡선발생기의 구성을 그림 11(a)에 나타내었다.  $P_{i,j}$ 가 순서적으로 입력되므로 기본 처리요소 입력단에 두개의 레지스터를 사용하며, 1 스텝에서 연산된  $P_{01}$ 과 2스텝에서 연산된  $P_{12}$ 가 출력단의 레지스터에 저장되고 3스텝에서

는 MUX를 사용하여  $P_{01}$ 과  $P_{02}$ 를 선택하며 연산 후 곡선 구성점  $P_{12}$ 를 생성한다.

그림 11(b)는 그림 5(b)와 같이 열방향으로 출력을 생성하고자 하는 경우 그림 11(a)의  $PE_2$ 의 결과가 4스텝마다 발생하므로 4스텝마다 발생하는 제어신호  $cs$  및 MUX를 사용하여 순차적으로 출력할 수 있다.

2. B-스프라인 곡선 발생기의 배합함수( $N_{i,k}$ ) 및 기본 처리요소 설계

조정점 6개( $n=5$ )이고,  $K=3$ 인 경우 식(8)과 식(9)에 의해서 생성된 그림 7의 곡선 발생기에 대한 기본 처리요소 및  $N_{i,k}$  연산부를 그림 12에 나타내었다.

그림 12에서 식(8-1)과 식(8-2)의 제산연산은 쉬프트 레지스터 및 우측의 제어신호 생성부 출력  $cs_1$ 과  $cs_2$ 에 의해 수행되며,  $N_{i-1,3}$ 는  $N_{i-2,3}$ 와  $N_{i,3}$ 의 감산에 의하여 산출된다. 기본 처리요소에서는 연산된  $N_{i,k}$ 와  $P_i$ 의 승산 및 전단 기본 처리요소에서 연산된 결과와의 누산을 수행함으로써 곡선발생을 수행한다.

3. 행렬·벡터 연산기를 사용한 베지어 곡선 발생기의 기본 처리요소 설계

조정점 4개인 경우 행렬·벡터 연산용 시스템릭 어레이를 이용한 베지어 곡선 발생기의 기본 처리요소를 그림 13에 나타내었다. 그림 13(a)는 식(9)의  $G \cdot M_b$ 연산을 위한 행렬 벡터 연산용 시스템릭 어레이의 기본 처리요소이며, 승산 및 가산에 의한 결과를 누적하는 레지스터로 구성된다. 그림 13(b)는  $G \cdot M_b$ 연산시 출력된 벡터와 U벡터와의 연산을 위한 벡터·벡터 연산용 어레이의 기본 처리요소이며, 승산기 및 전단 기본 처리요소의 결과를 누적하는 가산기

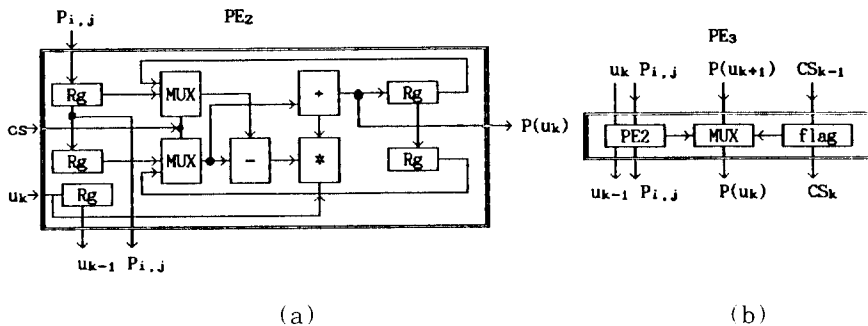


그림 11. 1차원 베지어 곡선 발생기의 기본 처리요소 (조정점 = 3)

(a) 2차원 발생기용 기본 처리요소 ( $PE_2$ ) (b) 1차원 발생기용 기본 처리요소 ( $PE_3$ )

Fig. 11. PE of bezier curve generator ( control points = 3 )

(a) PE for 2-dimensional generator (b) PE for 1-dimensional generator.

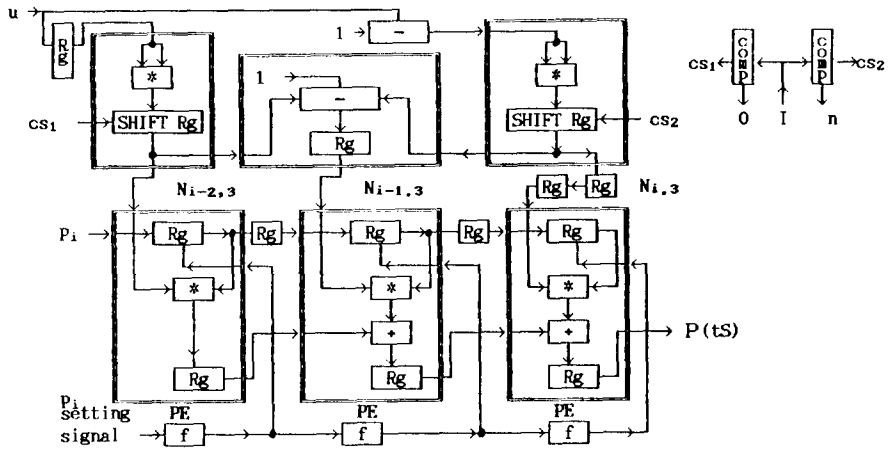


그림 12. B-스프라인 곡선 발생기의 내부구성도 (K=3, comp:비교기, f : 플랙 Rg.)

Fig. 12. Internal structure of B-spline curve generator. (k=3, comp:comparator, f: flag Rg.).

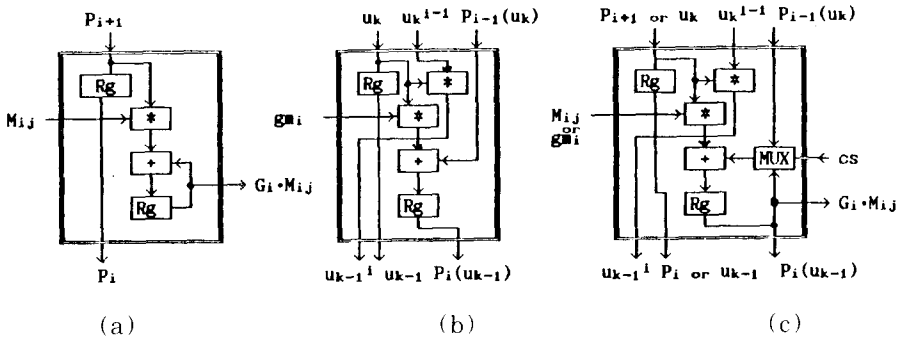


그림 13. 행렬 벡터 연산용 시스톨릭 어레이를 이용한 곡선 발생기의 기본 처리요소

(a) 행렬 벡터 연산용 시스톨릭 어레이의 기본 처리요소 (b) 벡터 벡터 연산용 시스톨릭 어레이의 기본 처리요소 (c) 행렬 벡터 연산 및 벡터 벡터 연산용 시스톨릭 어레이의 기본 처리요소

Fig 13. PE of curve generator using systolic array for Matrix Vector operation

(a) PE of Systolic Array for Matrix Vector operation (b) PE of Systolic Array for Vector Vector operation (C) PE of Systolic Array for Matrix Vector operation and Vector Vector operation.

로 구성된다.

그림 13 (a)는 연산된 결과를 누적하는 구조이며, 그림 13 (b)는 연산된 결과를 다음 기본 처리요소에 전달하는 구조로 되어 있으므로, 연산기는 동일하며 연산결과물의 전달 경로만 다르다. 그러므로 그림 13 (c)와 같이 MUX를 사용하여 연산결과물의 경로를 선택할 수 있는 구조로 기본 처리요소로 설계함으로써, 그림 9에 설계한 1개의 행렬·벡터 연산 및 벡터·벡터 연산용 시스톨릭 어레이만을 사용한 곡선발생기의 기본 처리요소로 사용할 수 있다.

### VII. 결과 및 고찰

본 논문에서 설계한 곡선 발생기의 논리적인 검증 을 위하여 C언어를 사용하여 시뮬레이션하였다. 매 사이클마다 설계된 곡선 발생기의 기본 처리요소에서 수행되는 연산과정을 공간-시간(사이클) 도표(Space - Time diagram)를 사용하여 그림 14에 표시하였으며, 곡선 구성점이 발생하는 부분을 중선으로 표시 하였다.

곡선 발생기에서 가산기의 연산시간을  $t_{add}$ , 승산기



$s^T$	PE( $P_{01}$ )	PE( $P_{12}$ )	PE( $P_{23}$ )	PE( $P_{02}$ )	PE( $P_{13}$ )	PE( $P_{03}$ )
1	$P_{01}(u_1)$	$P_{12}(u_1)$	$P_{23}(u_1)$	×	×	×
2	$P_{01}(u_2)$	$P_{12}(u_2)$	$P_{23}(u_2)$	$P_{02}(u_2)$	$P_{13}(u_2)$	×
3	$P_{01}(u_3)$	$P_{12}(u_3)$	$P_{23}(u_3)$	$P_{02}(u_3)$	$P_{13}(u_3)$	$P_{03}(u_3)$
4	$P_{01}(u_4)$	$P_{12}(u_4)$	$P_{23}(u_4)$	$P_{02}(u_4)$	$P_{13}(u_4)$	$P_{03}(u_4)$
:	:	:	:	:	:	:

(a)

$s^T$	1	2	3	4	5	6	7	8	9	10	11
PE	×	$P_{01}$	$P_{12}$	$P_{23}$	×	$P_{01}$	$P_{12}$	$P_{23}$	×	$P_{01}$	...
PE	×	×	×	$P_{02}$	$P_{13}$	×	×	$P_{02}$	$P_{13}$	×	...
PE	×	×	×	×	×	$P_{03}$	×	×	×	$P_{03}$	...

(b)

$s^T$	1	2	3	4	5	6	7	8	9	10	11	...
PE	×	$P_{01}$	$P_{12}$	$P_{02}$	×	$P_{01}$	$P_{12}$	$P_{02}$	×	$P_{01}$	$P_{12}$	...
PE	×	×	×	$P_{12}$	$P_{02}$	×	$P_{01}$	$P_{12}$	$P_{02}$	×	$P_{01}$	...
PE	×	×	×	$P_{01}$	$P_{12}$	$P_{02}$	×	$P_{01}$	$P_{12}$	$P_{02}$	×	...
PE	×	×	×	×	$P_{01}$	$P_{12}$	$P_{02}$	×	$P_{01}$	$P_{12}$	$P_{02}$	...

(c)

$s^T$	1	2	3	4	5	...
$N_{i-2,3}$	$N_{i-2,3}(t_1)$	$N_{i-2,3}(t_2)$	$N_{i-2,3}(t_3)$	$N_{i-2,3}(t_4)$	$N_{i-2,3}(t_5)$	...
$N_{i-1,3}$	×	$N_{i-1,3}(t_1)$	$N_{i-1,3}(t_2)$	$N_{i-1,3}(t_3)$	$N_{i-1,3}(t_4)$	...
$N_{i,3}$	×	×	$N_{i,3}(t_1)$	$N_{i,3}(t_2)$	$N_{i,3}(t_3)$	...
PE	×	$P_1(t_1)$	$P_1(t_2)$	$P_1(t_3)$	$P_1(t_4)$	...
PE	×	×	$P_2(t_1)$	$P_2(t_2)$	$P_2(t_3)$	...
PE	×	×	×	$P_3(t_1)$	$P_3(t_2)$	...

(d)

$s^T$	1	2	3	4	5	6	7	8	9	10	11	12	13	...
PE	×	×	×	$GMx_1$	×	×	×	$GMy_1$	$Px_1(u_1)$	$Py_1(u_1)$	$Px_1(u_2)$	$Py_1(u_2)$	$Px_1(u_3)$	...
PE	×	×	×	×	$GMx_2$	×	×	$GMy_2$	$Px_2(u_1)$	$Py_2(u_1)$	$Px_2(u_2)$	$Py_2(u_2)$	$Px_2(u_3)$	...
PE	×	×	×	×	×	$GMx_3$	×	×	$GMy_3$	$Px_3(u_1)$	$Py_3(u_1)$	$Px_3(u_2)$	$Py_3(u_2)$	...
PE	×	×	×	×	×	×	$GMx_4$	×	×	×	$GMy_4$	$Px_4(u_1)$	$Py_4(u_1)$	...

(e)

그림 14. 곡선 발생기 연산과정에 대한 공간-시간 도표 (s:공간, t:시간)

(a) 2차원 어레이로 구성된 베지어 곡선발생기 (조정점=4) (b) 1차원 어레이로 구성된 베지어 곡선 발생기 (조정점=4) (c) 두번 사상된 PE를 사용한 베지어 곡선 발생기(조정점=3) (d) B-스프라인 곡선 발생기(k=3, 조정점=6) (e) 행렬 벡터 연산용 어레이 프로세서를 이용한 베지어 곡선 발생기(조정점=4)

Fig 14. Space-Time diagram of Curve generator operation (s:space, t:time)

(a) Bezier curve generator using 2-d Array(control points=4) (b) Bezier curve generator using 1-d Array(control points=4) (c) Bezier curve generator using double mapped PE (control points=3) (d) B-spline curve generator(k=3, control points=6) (e) Bezier curve generator using systolic array for Matrix Vector operation (control points=4).

의 연산 시간을  $t_{mul}$ , 감산기의 연산 시간을  $t_{sub}$ 이라 할 때, 기본연산시간  $t_{clock}$ 을 기본 처리요소의 연산 시간이라 하면  $t_{clock} = t_{pe}$  이며, 연산기의 연산시간

$t_{top}$  일 때,  $t_{top} = \max \{ t_{add}, t_{mul}, t_{sub} \}$  이다. 기본처리 요소의 연산시간은 행렬 벡터 연산용 시스틀릭 어레이 프로세서의 경우는  $t_{pe} = 2 * t_{top}$ 이며,

이외의 제안된 곡선 발생기의 경우는  $tpe = 3 * top$ 이다. 참고문헌 [5]의 곡선 발생기의 경우는  $tpe = 4 * top$ 이며, 참고문헌 [6]의 베지어 곡선 발생기의 경우는  $tpe = 9 * top$ 이며, B-스프라인 곡선 발생기의 경우는  $tpe = 6 * top$ 이다. 그림 14의 기본 처리요소의 연산시간 및 공간-시간 도표에 의하여 산출된 곡선 발생기의 연산 시간 및 곡선 발생기에 소요된 연산기 수의 비교를 표 1에 나타내었다. 생성할 곡선이 2차원 곡선이므로 행렬 벡터 연산용 시스톨릭 어레이 프로세서를 이용한 곡선 발생기를 제외한 곡선 발생기는 기본 처리요소가 2배로 증가하며, 소요된 연산기의 수도 2배가 된다. 표 1에서 행렬·벡터 연산용 시스톨릭 어레이 프로세서를 이용한 경우가 잠재시간은 많지만 연산시간 및 연산기 사용수에 있어서 가장 우수함을 알 수 있다. 또한 조정점이 3개인 경우 그림 4(c)의 PE를 사용한 베지어 곡선 발생기는 연속적인 곡선이 발생될 수는 있지만 2차원 발생기와 비교하여 볼 때 연산기 사용수에 있어서 효율성이 떨어짐을 알 수 있다

표 1. 곡선 발생기의 연산시간 및 연산기 수 비교(m : 곡선 구성점의 수)

Table 1. The Comparison of operation time anber of operators(m : No. of generated curve points).

곡 선 발 생 기 종 류	연산시간(top)	소요된 연산기 수				
		*	-	+	합	
조정점 = 4 이 어	제안된 2차원 어레이	3m + 6	12	-	12	36
	1차원 제안된 어레이	12m + 6	6	-	6	18
	어레이 참고문헌 [6]	9m + 18	6	-	6	18
	제안된 행렬·벡터 연산용 어레이	2m + 22	4	-	4	8
조정점 = 3 곡 선	제안된 2차원 어레이	3m + 3	6	-	6	18
	1차원 제안된 어레이 (7점4(c) PE사용)	3m + 9	8	-	8	24
	어레이 참고문헌 [6]	9m + 9	4	-	4	12
	제안된 어레이	3m + 9	8	2	4	14
B-스프라인 곡 선	제안된 어레이	4m + 12	8	-	8	16
	참고문헌 [5] 1차원 어레이	4m + 8	12	12	12	48
	참고문헌 [6] 2차원 어레이	5m + 12	24	14	18	28

B-스프라인 곡선 발생기는 행렬·벡터 연산용 시스톨릭 어레이 프로세서를 이용한 곡선 발생기에 비하여 연산속도는 늦지만 조정점의 수가 많고, 유연성 있는 곡선발생을 필요로 하는 경우에는 적용 가능하다.

Ⅶ. 결 론

고속의 곡선 발생기는 컴퓨터를 이용한 설계 및 시뮬레이션등과 같이 실시간 그래픽 처리가 요구되는 분야에서 필수적인 요소이다.

그러므로, 본 논문에서는 곡선 근사화 방법중 대표적인 베지어 곡선 및 B-스프라인 곡선을 고속으로 생

성하기 위한 곡선 발생기를 설계하였다. 베지어 곡선 발생기는 베지어 곡선의 행렬 표현식 및 곡선 발생식의 순환성을 사용하여 두가지 방법으로 설계하였다. 곡선 발생식의 순환성을 사용하는 경우, 조정점이 4개일 때, 2차원 어레이 구조를 갖는 베지어 곡선 발생기와 1차원 어레이 구조의 베지어 곡선 발생기를 설계하였으나, 1차원 어레이로 설계한 경우 출력의 연속성을 갖지 못하는 단점을 갖는다. 그러므로 조정점이 3개인 경우, 2차원 곡선 발생기를 두번 사상한 기본 처리요소를 사용하여 연속적인 출력이 가능한 1차원 곡선 발생기를 설계하였다. 2차원 곡선인 경우 곡선의 구성점( $P_x, P_y$ )을 생성하기 위하여 두배의 기본 처리요소가 필요하므로 소요된 연산기 수도 2배로 증가된다. 그러므로 행렬 표현식을 이용한 곡선 발생기는 x,y 위치성분에 대한 결과를 레지스터에 저장하고, 시스톨릭 어레이의 기본 처리요소를 행렬·벡터 연산 및 벡터·벡터 연산이 가능하도록 설계하여 기본 처리요소의 사용수가 감소되도록 설계하였다. B-스프라인 곡선 발생기는 조정점이 6개이고 매듭 k=3인 경우, 배합함수에 대한 의존성 및 수식의 간략화를 사용하여 배합함수 연산부의 크기가 감소되었으며, 조정점과 배합함수와의 연산부는 1차원 시스톨릭 어레이 구조로 설계하였다.

설계된 곡선 발생기와 참고문헌 [5], [6]의 곡선 발생기의 비교시 연산속도 및 소요연산기 수에 있어서 행렬·벡터 연산용 시스톨릭 어레이 프로세서를 이용한 곡선 발생기 효율적임을 알 수 있다. 또한 조정점의 수가 많고 유연성 있는 곡선발생을 필요로 하는 경우에는 B-스프라인 곡선 발생기가 유효함을 알 수 있었다. 앞으로 연구결과를 실용화하기 위하여 본 연구에서 설계된 곡선 발생기를 이용한 그래픽 프로세서의 설계에 관한 연구가 계속 되어야할 것이다.

參 考 文 獻

[1] Foley, Van Dam, Feiner, and Hugles, Computer Graphic Princilpe and Practice, Addison- Wesley, Second Edition, 1990.

[2] Ph. Coueignoux, "Character Generation by Computer", Computer Vision, Graphics and Image Processing, Academic Press, Vol 16, No.3, pp 240-269, July, 1981.

[3] William.M.Neuman, Robert F. Sproull, Principle of Interactive

- Computer Graphics, Mc Graw-hill Inc., Second Edition, 1986.
- [4] Theo Pavlidis, Algorithm for Graphics and Image Processing, Computer Science Press, 1982.
- [5] Chi-Hsu Wang, Hore-Yuan, and Rong-Sheng Wen, "Pipelined Computations of B-spline Curve", *IEEE Trans. on Systems, Man, and Cybernetics*, Vol 22, No. 2, pp. 327-331, March/April, 1992.
- [6] 金龍成, 趙源敬, "고속 곡선 근사화를 위한 1차원 시스틀릭 어레이 프로세서 설계에 관한 연구", 慶熙大學校 電子工學科 電子通信工學研究 論文集, 第10卷, 第 1號, pp. 25-28, 3月, 1991.
- [7] 李泰炯, 黃奎哲, 李潤泰, 裴鐘洪, 慶宗旻, "한글 외과선 생성을 위한 고속 프로세서의 설계", 電子工學會論文誌, 第 28卷, B編, 第 9號, pp.21-27, 9月, 1991
- [8] 金龍成, 趙源敬, "행렬 벡터 연산용 1-차원 시스틀릭 어레이 프로세서를 이용한 그래픽 가속기의 설계", 電子工學會論文誌, 第 30卷, B編, 第 1號, pp. 1-9, 1月, 1993.
- [8] Tsuneo Ikedo, "High-Speed Techniques for a 3-D Color Graphics Terminal", *IEEE Computer Graphics and Application*, Vol 4, No. 5, pp. 46-58, May, 1984.
- [9] Vincent J. Harrand, John P. Ziebearth, and Amar Choudry, "Fast subdivision of Bezier surfaces", *Simulation*, Vol 54, No. 5, pp. 241-244, May, 1990.
- [10] J.A.K.S. Jayassinghe, F.Moelaert Eihadidy, G. Karagiannis, Otto E. Herrmann, and J. Smit, "Two-Level Pipelined Systolic Array Graphics Engine", *IEEE Journal of Solid-State Circuits*, Vol 26, No.3, pp. 229-236, March, 1991.
- [11] S.Y.Kung, VLSI Array Processors, Prentice-Hall, 1988.

---

 著 者 紹 介
 

---

金 龍 成(正會員) 第 30卷 B編 第 1號 參照  
 현재 경희대학교 전자공학과 박사  
 과정 재학중

趙 源 敬(正會員) 第 30卷 B編 第 1號 參照  
 현재 경희대학교 전자공학과 교수