

소프트웨어 펄스 보간기의 성능비교

안 중 환*, 임 현 석**, 이 원 규***

Study on the Performance Comparision of Software Pulse Interpolators

J.H.Ahn*, H.S.Lim**, W.K.Lee***

ABSTRACT

Interpolator is a very important element in NC machines in that it controls tool path and speed. In this paper, studied were extensive interpolation characteristics of reference pulse method among various interpolation and pulse generation methods. Specifically, processing speed and path error of DDA, SPD and SFG methods were compared and analyzed against line, circle and ellipse. As a result, in the point of processing speed, SPD method was found to be the best for line interpolation, SFG method for circle and ellipse, and DDA method was found to be the slowest for all paths. In the point of path error, DDA method was found to have the biggest error for all kinds of paths.

Key Words : Pulse Interpolator(펄스보간기), NC Machine(수치제어기계), Reference Pulse(기준펄스), Pulse generation(펄스생성), DDA(Digital Differential Analyser), SPD(Smooth Pulse Distribution), SFG(Saita Function Generator)

1. 서론

보간기(Interpolator)는 NC 공작기계에서 공구의 궤적 및 이송 속도를 제어하는 중요한 요소이다. 보간기는 보간계산과 펄스의 발생을 전자회로로 수행하는 하드웨어 보간기와 컴퓨터 프로그램으로 수행하는 소프트웨어 보간기로 나눌 수 있는데, 하드웨어 보간기는 고속보간이 가능한 반면 유연성이 부족한 단점이 있기 때문에 공작기계를 CNC화 한 뒤로는 그 의존도가 점점 낮아지고 있다.^(1,2)

소프트웨어 보간기에 사용되는 방법에는 기준펄스

(Reference Pulse) 방법과 샘플치(Sampled Data) 방법이 있다. 기준펄스 방법은 컴퓨터가 외부 동기신호에 맞추어서 계산을 수행하고 그 결과에 따라 해당 축에 직접 펄스를 발생시켜 축을 구동시킨다. 샘플치 방법은 하드웨어 보간기와 컴퓨터를 함께 이용한 것으로서, CPU는 궤적을 허용오차 내에 들도록 근사화 하고 하드웨어 보간기는 일정주기로 근사직선 정보를 해당 축에 출력시켜 축을 구동시킨다.

기준펄스 방법은 컴퓨터가 보간계산을 하여 직접 펄스를 내므로 CPU의 처리 속도에 따라 이송 속도가 제한을 받지만 샘플치 방법보다 정밀도 높은 가공을 가능하게 한

* 부산대학교 정밀기계공학과, 기계기술연구소

** 현대정공

*** 울산대학교 생산기계공학과

다. 샘플치 방법은 처리 속도로 인한 속도 제한이 없기 때문에 고속가공용 보간기에 적합하지만, 근사직선에 의한 방법이므로, 기준펄스 방법보다 궤적오차가 크고 제어 루프가 복잡하며 근사직선에 대한 정보를 저장할 많은 내부 메모리를 필요로 한다는 단점이 있다.^(3,4) Y.Koren은 DDA(Digital Differential Analyzer) 방식, 대수연산(Stairs Approximation) 방식, Direct Search 방식의 특성을 알아보기 위하여, 원호보간을 대상으로 어셈블리어를 사용해 프로그래밍하여 각각의 경로오차, 속도의 균일성 등의 측면에서 비교, 분석하였다.⁽³⁾

본 연구에서는 더욱 광범위한 기준펄스 보간기의 특성을 조사하기 위하여 DDA^(2,3,5-8), SPD (Smooth Pulse Distribution)⁽⁶⁾, SFG(Saita Function Generator)^(6,9) 방식의 직선, 원호, 타원보간 성능을 알고리즘의 처리 속도, 경로오차의 측면에서 비교, 분석하였다.

2. 소프트웨어 펄스 보간의 원리

2.1 DDA 방식^(2,3,5-8)

DDA 방식은 수치 적분의 원리를 이용한 것으로서 Fig. 1과 같이 속도의 함수 $V(t)$ 가 주어질 때 이동거리 $S(t)$ 를 곡선의 아래 부분에 있는 미소 직사각형의 면적의 합으로 근사화 한다.

$$S(t) = \int_0^t V(t)dt \approx \sum_{i=0}^{k-1} V_i \Delta t \quad (1)$$

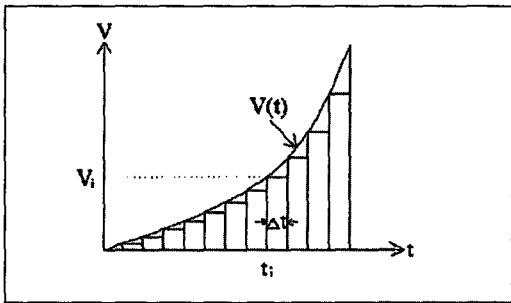


Fig.1 Digitized velocity integration of DDA method

i) 직선일 경우

원점으로부터 끝점 $P_e(x_e, y_e)$ 까지 속도 V 로 이동하므로 각 축의 위치는 다음과 같다.

$$\begin{aligned} x(t) \approx x_k &= \sum_{i=0}^{k-1} V_{xi} \Delta t = \sum_{i=0}^{k-1} V \Delta t (x_e / L) \\ y(t) \approx y_k &= \sum_{i=0}^{k-1} V_{yi} \Delta t = \sum_{i=0}^{k-1} V \Delta t (y_e / L) \end{aligned} \quad (2)$$

여기서 $L = \sqrt{x_e^2 + y_e^2}$ 이다.

ii) 원일 경우

반경이 R , 각속도 ω , 각 축방향의 속도는 $V_x(t) = -R \omega \sin(\omega t)$, $V_y(t) = R \omega \cos(\omega t)$ 이므로, 각 축의 위치는 다음과 같다.

$$\begin{aligned} x(t) \approx x_k &= x_s - \sum_{i=1}^{k-1} V_{xi} \Delta t = x_s - \omega \sum_{i=0}^{k-1} y_i \Delta t \\ y(t) \approx y_k &= y_s + \sum_{i=1}^{k-1} V_{yi} \Delta t = y_s + \omega \sum_{i=0}^{k-1} x_i \Delta t \end{aligned} \quad (3)$$

iii) 타원일 경우

장축 $2a$, 단축 $2b$, 각속도 ω , 각 축의 속도는 $V_x(t) = -a \omega \sin(\omega t)$, $V_y(t) = b \omega \cos(\omega t)$ 이므로, 각 축의 위치는 다음과 같다.

$$\begin{aligned} x(t) \approx x_k &= x_s - \sum_{i=1}^{k-1} V_{xi} \Delta t = x_s - (a/b) \omega \sum_{i=1}^{k-1} y_i \Delta t \\ y(t) \approx y_k &= y_s - \sum_{i=1}^{k-1} V_{yi} \Delta t = y_s - (a/b) \omega \sum_{i=1}^{k-1} x_i \Delta t \end{aligned} \quad (4)$$

여기서 x_s 및 y_s 는 시작점의 x 및 y 좌표값이다.

근사 요소의 갯수(반복횟수)를 N , 전체 근사보간 시간을 T 라고 하면 $\Delta t = T/N$ 이다. NC공작기계에서 공구 및 테이블의 이송은 1 BLU 단위로 이루어지므로 계산상의 단위시간 (Δt) 당 이동거리는 1 BLU 값 보다 작아야만 한다는 조건으로부터 N 값이 결정된다.

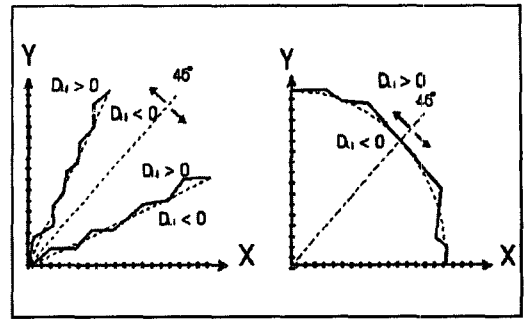
i) 직선일 경우, 단위 시간당 각 축의 이동 거리는 식 (2)로부터

$$\begin{aligned} \Delta x &= V_{x_e} \Delta t / L = x_e \Delta t / T = x_e / N \\ \Delta y &= V_{y_e} \Delta t / L = y_e \Delta t / T = y_e / N \text{ 이고,} \\ \max(\Delta x, \Delta y) &\leq 1 \text{ 이어야 하므로} \\ N &\geq \max(x_e, y_e) \text{ 이다.} \end{aligned}$$

ii) 원일 경우, 식 (3)으로부터

$$\Delta x = \omega y_i \Delta t = \omega y_i T / N = \theta y_i / N \leq 1,$$

$\Delta y = \omega_{x_i} \Delta t = \omega_{x_i} T/N = \theta_{x_i}/N \leq 1$ 이고,
 $\max[\theta_{x_i}, \theta_{y_i}] \leq 1$ 이어야 하므로
 $N \geq \max[\theta_{x_i}, \theta_{y_i}]$ 이다.
 iii) 타원일 경우, 식 (4)로부터
 $\Delta x = (a/b)\omega_{y_i} \Delta t = (a/b)\omega_{y_i} T/N$
 $= (a/b)\theta_{y_i} \leq 1,$
 $\Delta y = (b/a)\omega_{x_i} \Delta t = (b/a)\omega_{x_i} T/N$
 $= (b/a)\theta_{x_i} \leq 1$ 이고,
 $\max[(b/a)\theta_{x_i}, (a/b)\theta_{y_i}] \leq 1$ 이어야 하므로
 $N \geq \max[(b/a)\theta_{x_i}, (a/b)\theta_{y_i}]$ 이다.



(a) line (b) circular arc

Fig. 3 Pulse interpolation by SPD method

Table 1. Interpolation rule of SPD method

	기울기 < 45°		기울기 ≥ 45°		반복 횟수
직선	D _{ij} ≥ 0	D _{ij} < 0	D _{ij} ≥ 0	D _{ij} < 0	max(dx, dy)
선	+X	+X, +Y	+X, +Y	+Y	
원호	D _{ij} > 0	D _{ij} ≤ 0	D _{ij} > 0	D _{ij} ≤ 0	$\sqrt{\{(dx)^2 + (dy)^2\}}$
타원	D _{ij} > 0	D _{ij} ≤ 0	D _{ij} > 0	D _{ij} ≤ 0	"
원	-X	-X, +Y	-X, +Y	+Y	

D_{ij} : (i, j) 점에서의 판별식 값

$dx = x_e - x_s, dy = y_e - y_s$

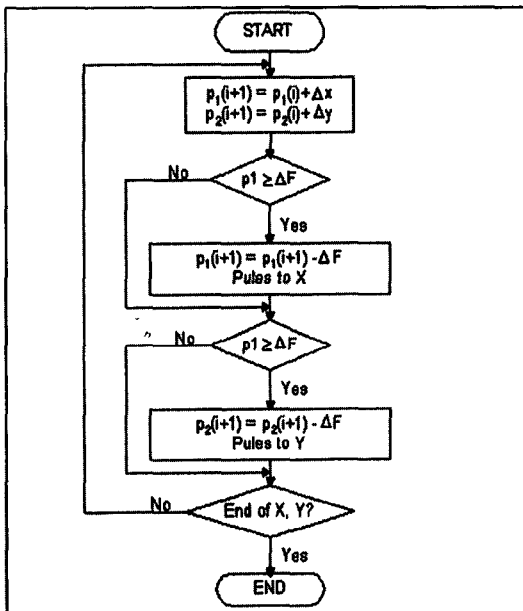


Fig.2 Flow chart of DDA linear interpolator

2.2 SPD 방식⁽⁸⁾

SPD 방식은 원활한 궤적의 보간을 위하여 대수연산 방식을 개선한 것으로, 대수연산 방식에서는 사용되지 않는 45° 방향의 펄스를 내는 것이 그 특징이다. 이 것은 Fig. 3와 같이 임의의 함수에 대하여 그 점선의 기울기가 45° 보다 큰 영역과 작은 영역에 대해 판별식을 이용하여 보간을 실행하는 방식이다. 1상환에 있는 직선, 원호 및 타원의 궤적에 대해 보간할 경우, 기울기의 영역과 판별식의 범위에 따라 발생시킬 각 축방향 펄스를 도표로 정리하면 Table 1과 같다.

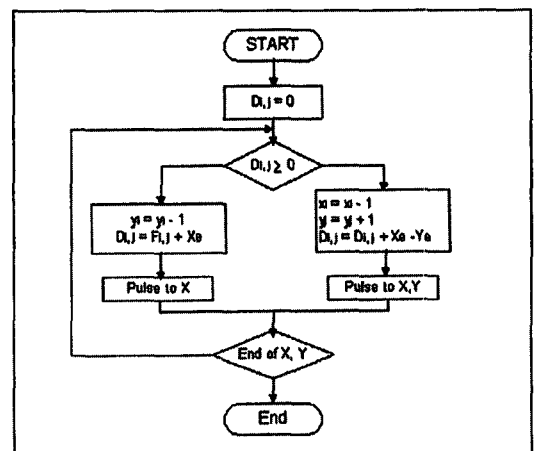


Fig. 4 Flow chart of SPD linear interpolator (for slope < 45°)

2.3 SFG 방식^(8,9)

임의의 두점을 보간할 할 때 각 축의 펄스를 내는 시간

을 조절함에 따라서 보간의 형태가 달라지게 된다. x축에 $a, a+\alpha, a+2\alpha, \dots, a+\alpha(x-1)$, y축에 $b, b+\beta, b+2\beta, \dots, b+\beta(y-1)$ 의 시간간격으로 각각 x, y개의 펄스를 발생시킬때 펄스를 발생하는 동안 걸리는 시간을 t라 한다면 다음과 같은 식이 된다.

$$\begin{aligned}
 t &= a + (a - \alpha) + (a - 2\alpha) + \dots + (a - \alpha(x-1)) \\
 &= ax - \alpha x(x-1)/2 \cong ax - \alpha x^2/2 \\
 t &= b + (b + \beta) + (b + 2\beta) + \dots + (b + \beta(y-1)) \\
 &= by + \beta y(y-1)/2 \cong by + \beta y^2/2
 \end{aligned} \tag{5}$$

위 식에서 t를 소거하면

$$ax - \alpha x^2/2 = by + \beta y^2/2 \tag{6}$$

이다.

(6)식에서

i) $\alpha=\beta=0$ 이면

$ax=by$ 인 직선의 식,

ii) $\alpha=-1, \beta=1$ 이면

$(x-a)^2 + (y+b)^2 = a^2 + b^2$ 인 원의 식,

iii) $\alpha=-1, \beta$ 가 임의의 값이면

$(x+a)^2 + \alpha(y+b/\beta)^2 = a^2 + \alpha b^2/\beta^2$ 인 타원의 식이 된다.

타원 보간에서 장축의 길이가 A이고 단축의 길이가 B 인 경우 $\alpha=(A/B)^2$ 이다.

Fig. 5 (a), (b)는 각각 직선과 원호의 펄스열을 나타내는 그림이다. SFG 방식에서는 펄스 발생 주기가 짧은 축을 기준축으로 두는데, 직선의 경우는 기준축이 일정하지만, 원호나 타원의 경우는 위치에 따라 기준축이 바뀐다. 원호 보간의 경우 45° 를 경계로 하여 45° 보다 작은 영역에서는 Y축이, 45° 보다 큰 영역에서는 X축이 기준

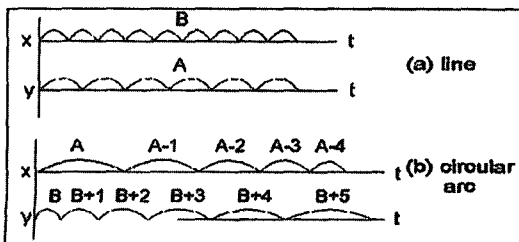


Fig. 5 Pulse interpolation of SFG method

축이 되고, 타원의 경우 $A^2y=B^2x$ 가 경계가 된다. 기준축에서는 매번 펄스를 내고 상대축에서 펄스가 발생할 요건이 만족되면 기준축에 동기하여 펄스를 낸다.

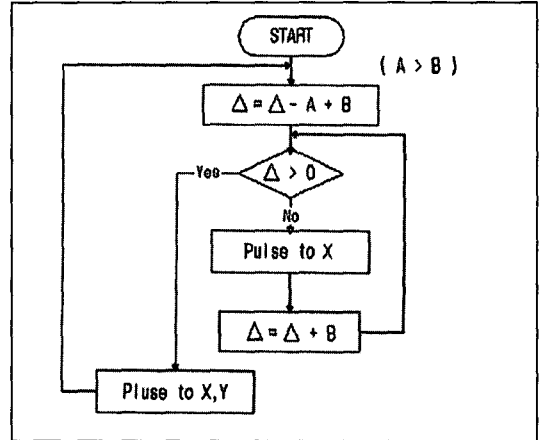
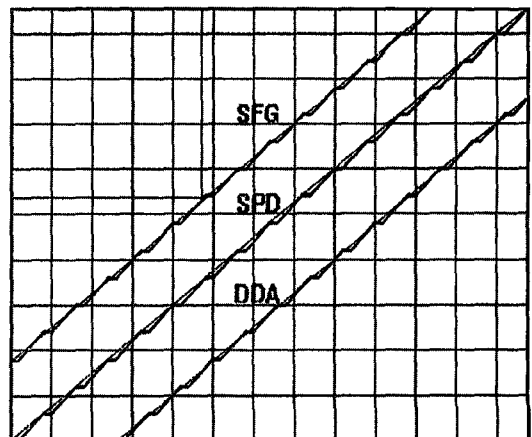


Fig. 6 Flow chart of SFG linear interpolator

3. 성능 비교 실험 및 결과

3.1 경로 보간 시뮬레이션

직선, 원, 타원을 대상 경로로 하여 각 보간 방식의 경로 보간을 컴퓨터 화면상에서 시뮬레이션하였다. Fig. 7 (a), (b)에 각 보간방식의 원호, 직선, 타원에 대한 경로 보간특징이 잘 나타나고 있다. 직선 보간의 경우 맨 위가 SFG, 중간이 SPD, 맨 아래의 경우가 DDA 보간 방식이다. SFG 방식과 SPD 방식은 보간 형태가 매우 유사하



(a)Line

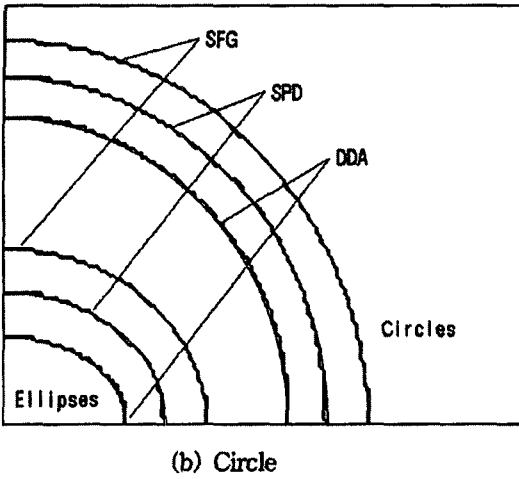


Fig. 7 Displayed interpolation results for circles and ellipses by three methods: DDA, SPD, SFG

다. 원호 및 타원 보간의 경우 가장 큰 원(또는 타원)이 SFG, 중간의 원(또는 타원)이 SPD, 가장 작은 원(또는 타원)이 DDA 방식으로 보간한 것이다.

3.2 지령경로 오차

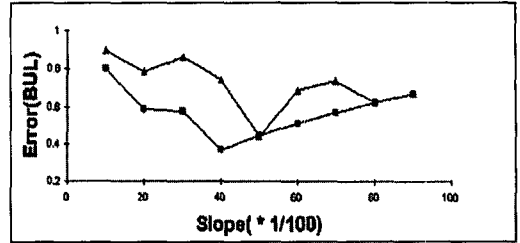
지령경로 오차의 표시는 SFG, SPD, DDA 방식에 대해서 각각 \blacklozenge , \blacksquare , \blacktriangle 으로 표시하였다.

1) 직선보간 오차

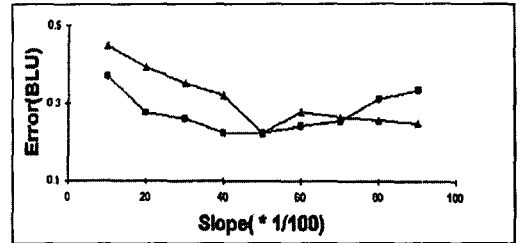
이것은 보간된 점에서 직선까지의 수직 방향의 오차를 나타낸다. X축의 길이를 100단위로 하고 Y축의 길이를 10단위에서 90단위까지 10단위씩 증가 시켰을 때, 3가지 보간 방법에 의해 발생된 직선보간 오차의 최대값과 평균값을 각각 Fig. 8 (a), (b)에 나타내었다. SFG, SPD 방식의 최대오차 및 평균오차는 같게 나타났다. 최대오차에서는 DDA 방식이 다른 두가지 방식보다 더 크게 나타났다. 평균오차에서는 0° 에서 35° 까지 DDA 방식이 더 크게 나타났고, 35° 에서 45° 까지는 SPD 및 SFG 방식이 더 크게 나타났다.

2) 원호 보간 오차

이것은 보간된 점과 원점간의 거리에서 반경을 뺀 오차를 나타낸다. 0° 에서 90° 까지의 원호에 대하여 보간하였으며, 반경을 10000단위에서 100000단위까지 10000단위씩 증가시켰을 때의 최대오차 및 평균오차를 Fig. 9 (a), (b)에 나타내었다. 원호보간에서의 오차는 DDA 방식이 가장 크고, SFG방식이 가장 작게 나타났다.

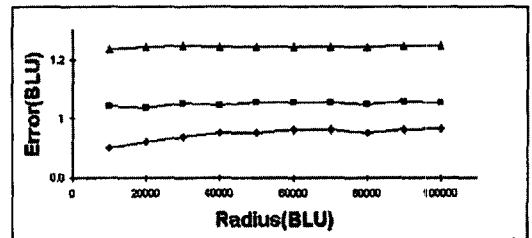


(a) Peak error

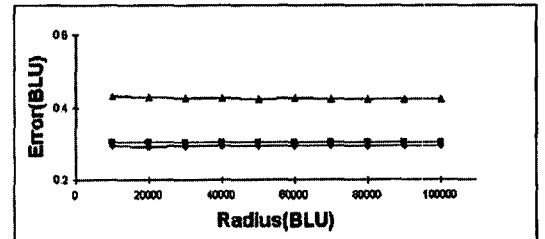


(b) Average error

Fig. 8 Command path errors for linear interpolation



(a) Peak error



(b) Average error

Fig. 9 Command path errors for circular interpolation

3) 타원보간 오차

보간점에서 타원까지 수직거리 계산은 까다롭기 때문에 Fig. 10 과 같은 근사오차를 타원보간 오차로 사용하였다. 장축과 단축의 비를 1/2로 두고 0° 에서 90° 까지 보간하였고, 장축의 길이를 10000단위부터 100000단위

까지 10000단위씩 증가시켰을 때의 최대오차 및 평균오차를 각각 Fig. 11 (a), (b)에 나타내었다. 타원보간에서의 오차도 DDA방식이 가장 크고, SFG방식이 가장 작게 나타났다.

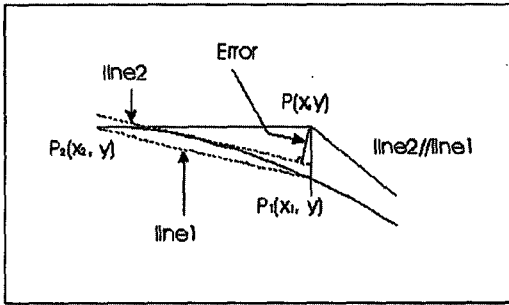
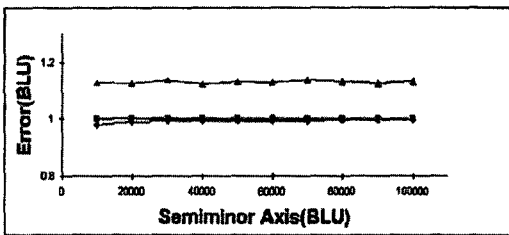
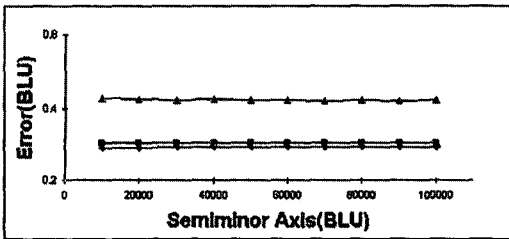


Fig. 10 Approximation of path error for ellipse



(a) Peak error



(b) Average error

Fig. 11 Command path errors for elliptic interpolation

3.3 알고리즘의 처리 속도

Turbo C 프로그래밍 언어를 사용하고 수치계산은 부동소수점으로 하여 보간 알고리즘을 구현한 후 각 알고리즘의 처리속도를 측정하였다. 처리속도의 측정을 위한 하드웨어 셋업은 Fig.12와 같이 컴퓨터와 스텝핑 모터 구동기를 연결시켜주는 인터페이스 회로 및 스텝핑 모터로 구동되는 x-y플로터로 구성된다(구체적인 회로는 부록 참조).

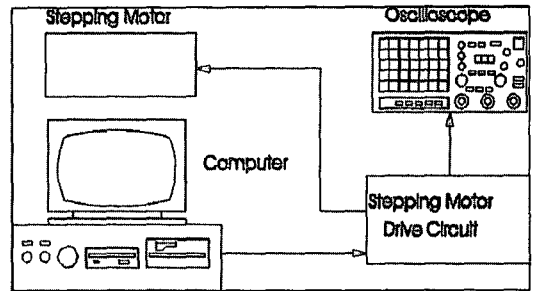


Fig.12 Hardware set-up for execution time measurement.

처리속도는 한번의 보간루프를 실행할 때 걸리는 시간으로 나타낼 수 있는데, 보간루프의 실행시간이 짧을수록 알고리즘의 처리속도는 빠르다. 측정방법은 IBM/AT 컴퓨터를 사용하여 보간루프를 실행하는 동안 임의의 출력포트를 High(또는 Low) 상태가 되게 하고, 그 다음의 보간루프를 실행하는 동안에는 Low(또는 High) 상태가 되도록 반전시켜 오실로스코프로 시간을 측정하였다. Fig.13은 한번의 보간루프 실행에 걸리는 시간을 측정한 한 예(SPD 직선보간)이다.

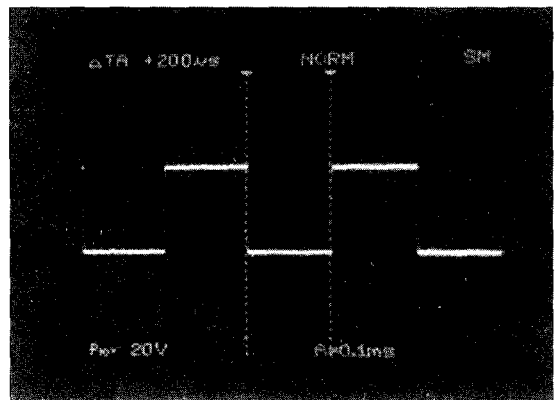


Fig. 13 An example of execution time measurement for one loop of a given algorithm(DDA linear interpolation)

측정 결과는 아래의 table. 2와 같다.

Table 2. Execution time of an interpolation loop(msec)

	SPD	SFG	DDA
직선	0.18	0.18	0.51
원호	0.51	0.38	0.62
타원	0.51	0.38	0.62

프로그램 실행 시간은 프로그램의 길이, 곱하기, 나누기, 비교문 등의 연산횟수가 많을 수록 길어진다. DDA 원호 보간에서 실행 시간이 많이 걸리는 것은 비교문과 곱하기 연산이 SPD, SFG 방식보다 상대적으로 많았기 때문이라 여겨진다.

실행 속도의 면에서 어셈블리어로 프로그램하면 Table 2. 보다 훨씬 짧은 시간내에 처리될 것이지만, 전체적인 시간은 비슷한 비율로 줄어든 것이다.

4. 결 론

직선, 원, 타원을 대상으로 DDA, SPD, SFG 방식에 대한 보간 성능을 연구한 결과 다음과 같은 결론을 얻었다.

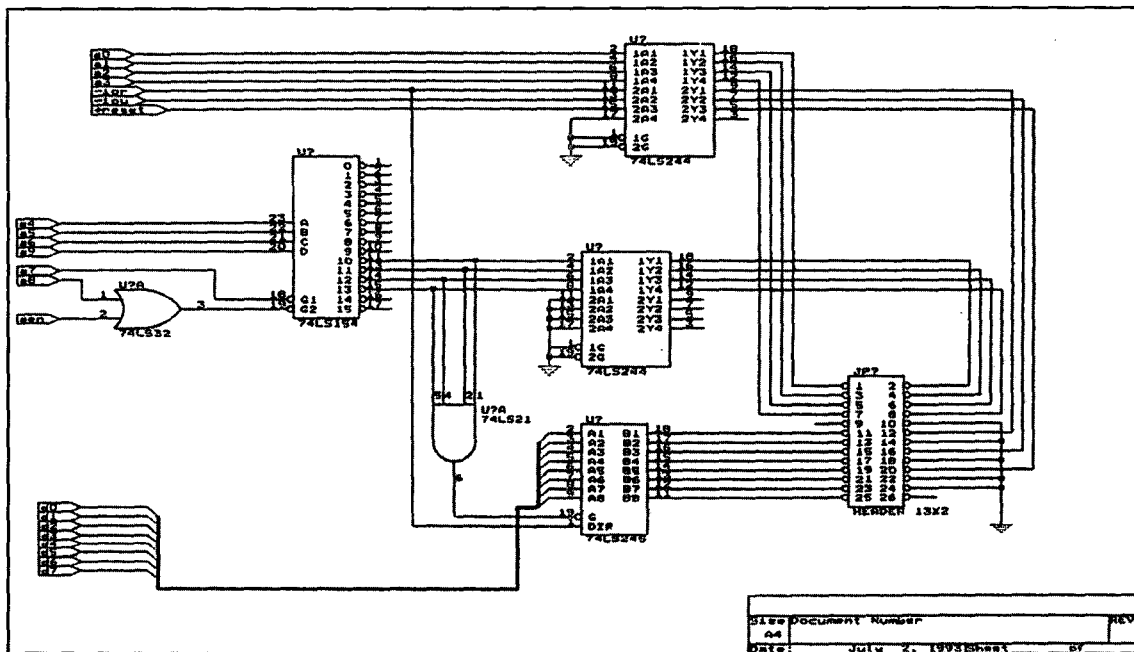
- 1) 직선, 원, 타원에 대하여 모두 보간 능력을 가지고 있다.
- 2) 처리 속도면에서는 직선 보간에서는 SPD 방식이, 원호 보간 및 타원 보간에서는 SFG 방식이 가장 빠르고, 모든면에서 DDA 방식이 가장 느리다.
- 3) 경로 오차는 직선 보간에서 35°에서 45°까지의 평균 오차를 제외하면 직선, 원호, 타원 보간의 모두에 대해서 DDA 방식이 가장 오차가 크다.

참 고 문 헌

1. Y.Koren, "Design of Computer Control for Manufacturing Systems", Trans. ASME. Journal of Engineering for Industry, Vol.101, pp. 326-331, August, 1979.
2. Y. Koren, Computer Control for Manufacturing Systems, McGraw-Hill., Inc., 1983.
3. O. Masory, Y.Koren, "Reference-Pulse Circular Interpolators for CNC Systems", Trans. ASME. Journal of Engineering for Industry, Vol.103, No. 1, pp. 131-136, February, 1981.
4. O.Masory, Y.Koren, "Reference-Word Circular Interpolators for CNC Systems", Trans. ASME. J. Engineering for Industry, Vol. 104, pp. 400-405, November, 1982.
5. M.Weak, Handbook of Machine Tools Vol. 3, John Wiley & Sons, 1984.
6. 김교형, 이기설, "2축 윤곽 제어를 위한 소프트웨어 보간자 개발에 관한 연구", 대한 기계학회 논문집, 제 21권, 제 2호, pp.389-396, 1988.
7. 최기봉, "2축 수치 제어용 보간기 개발과 위치 제어에 관한 연구", 한국 과학 기술원, 석사 학위 논문, 1987.
8. 大島康次郎, "位置決め 制御の現状とその應用", 機械の研究, 第32卷, 第11號, pp. 1239-1375, 1980.
9. 齊田伸雄, "新しい關數概念の創造", 安川電機, 第38卷, 第147號, No.3, pp.300-309, 1974.
10. Hong,Tao, "The High Precision DDA for Ellipse-Generation" NATO ASI Series Vol.F40, 1988

부 록

1. 컴퓨터와 구동기의 인터페이스 회로도



2. 스텝핑 모터 구동기 회로도

