

저가형 마이크로프로세서를 위한 연산처리 확장 모션제어 알고리즘

Motion Control Algorithm Expanding Arithmetic Operation for Low-Cost Microprocessor

문상찬, 김재준, 남규민, 김병수, 이순걸*

(Sang-Chan Moon¹, Jae-Jun Kim², Kyu-Min Nam³, Byoung-Soo Kim¹, and Soon-Geul Lee¹)

¹Kyung Hee University

²Korea Institute of Industrial Technology

³Sejong Industrial Co., Ltd.

Abstract: For precise motion control, S-curve velocity profile is generally used but it has disadvantage of relatively long calculation time for floating-point arithmetics. In this paper, we present a new generating method for velocity profile to reduce delay time of profile generation so that it overcomes such disadvantage and enhances the efficiency of precise motion control. In this approach, the velocity profile is designed based on the gamma correction expression that is generally used in image processing to obtain a smoother movement without any critical jerk. The proposed velocity profile is designed to support both T-curve and S-curve velocity profile. It can generate precise profile by adding an offset to the velocity profile with decimals under floating point that are not counted during gamma correction arithmetic operation. As a result, the operation time is saved and the efficiency is improved. The proposed method is compared with the existing method that generates velocity profile using ring buffer on a 8-bit low-cost MCU. The result shows that the proposed method has no delay in generating driving profile with good accuracy of each cycle velocity. The significance of the proposed method lies in reduction of the operation time without degrading the motion accuracy. Generated driving signal also shows to verify effectiveness of the proposed method.

Keywords: motor control, motion profile, motion control, MIPS (Million Instructions Per Second), micro processor, S-curve

I. 서론

최근의 산업사회에서 전자와 기계의 상호연관성으로 전자기계 기술 분야가 연구 및 산업 전반의 목적이 되고 있다. 특히 에너지 손실이나 소음 없는 고속 구동 등 이점이 많은 전동기개발에 많은 투자가 이뤄지고 있다[1]. 실제 산업현장에서 사용되는 모션 제어기술의 핵심은 성능에 의해 좌우되며 효과적인 제어를 위해 크게 속도형 또는 위치형으로 구분되어 진다. 제어기에서 아날로그 혹은 디지털 펄스열로서 제어를 하도록 하며 아날로그보다 정밀한 속도제어를 위하여 서보모터와 결합한 속도에 대한 되먹임 제어가 많이 사용되고 있다[2].

마이크로프로세서를 이용한 디지털 제어에서는 엔코더에서 발생하는 펄스열을 이용한 속도데이터를 사용하여 일정 시간동안 펄스열을 계수하고 간격을 감지하게 된다. 이때 기준속도를 설정하고 엔코더에서 발생하는 주파수 위상차

를 제어하는 PLL (Phase Locked Loop)제어를 사용하고 있다[3]. 마이크로프로세서에서 아날로그 데이터를 통해 정확한 속도제어 및 명령을 위하여 빠른 연산처리를 위해 정수연산과 고정소수점 연산을 사용한다. 이는 실수연산기 하드웨어가 비싸고, 웨이퍼 면적과 전력이 많이 소비되어 AVR이나 ARM 타입의 CPU에도 실수연산을 포함하지 않고 있기 때문이다. 그러나 보다 정확한 모션제어를 위해 실수연산이 필요하며 소프트웨어적인 연산을 통해 쉽고 다양한 모터 제어가 가능해야 된다.

본 논문에서는 상대적으로 느린 모터제어의 실수연산의 단점을 극복하고 실수연산 수준의 복잡한 처리를 가능하게 하는 새로운 형태의 프로파일 생성 방식을 제안하였다. 특히 메인 클럭의 주파수가 낮은 저가형태의 보드에서도 정확한 속도, 위치제어가 가능하도록 알고리즘을 구현함으로써 연산에 소요되는 시간을 단축함과 더불어 기존 제품의 성능개선에도 크게 도움을 줄 수 있음을 분석하였다.

제 II 장에서는 제안된 알고리즘의 설계방법을 설명하였으며, 제 III 장에서는 실제 적용된 알고리즘의 성능을 확인하고, 제 IV 장에서는 제안한 방식에 대한 평가와 결론을 맺었다.

II. 모션 제어 프로파일 성능 향상

1. 속도프로파일

* 책임저자(Corresponding Author)

논문접수: 2012. 7. 29., 수정: 2012. 10. 17., 채택확정: 2012. 10. 25.

문상찬, 김병수, 이순걸: 경희대학교 기계공학과

(msmoon83@gmail.com/poosoo@chol.com/sglee@khu.ac.kr)

김재준: 한국생산기술연구원(yellowseajun@hanmail.net)

남규민: 세종공업(kmzard@hotmail.com)

※ 본 논문은 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업(NIPA-2010-C1090-1021-0013)과 산업융합원천기술개발사업[10040990]에서 지원하여 연구하였음.

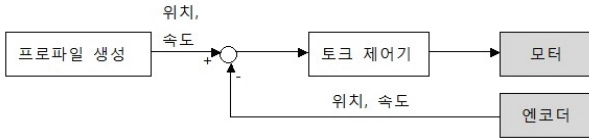


그림 1. 속도프로파일에 의한 모터제어기.
Fig. 1. Motor control with velocity profile.

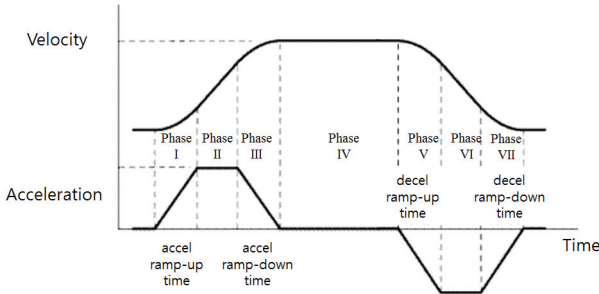


그림 2. S-커브형 가속 및 감속프로파일.
Fig. 2. S-curved accelerating & decelerating velocity profile.

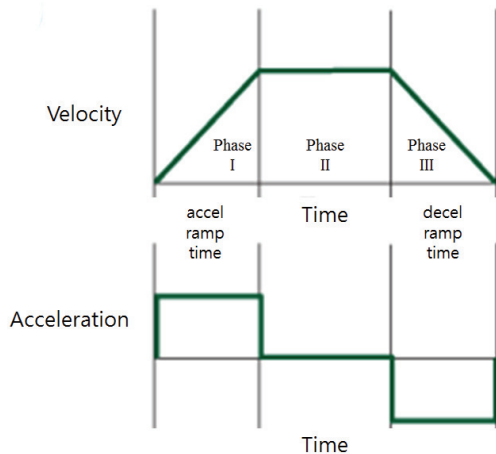


그림 3. 사다리꼴 가속속 프로파일.
Fig. 3. Trapezoidal accelerating and decelerating velocity profile.

기존의 모션 컨트롤에서는 원하는 성능을 얻기 위해 시스템 응답대신 모션 프로파일에 치중하게 된다. 이러한 비대칭 가속, S-커브 프로파일링은 기계가 빠르게 동작하기 위해 쓰이고 있다. 그림 1은 이와 같은 모션제어에 사용되는 하위 제어기에서의 프로파일 생성과 토크값 제어를 위한 피드백 제어 구조를 보여준다.

모션 프로파일의 목표점을 고려하여 프로파일을 최적화함으로써 ARM보드 타입의 모터구동 시스템의 성능에 많은 영향을 미친다. 그림 2와 같이 흔히 적용되어지는 S-커브 프로파일에서는 시작하여 최대 가속도에 도달하고 다시 0으로 가는 가속도의 선형적인 패턴이 두 번 반복되어 구간 I 부터 VII까지의 7단계로 구성되어 있음을 보여준다.

S-커브에서는 사다리꼴 프로파일에서 나타나는 jerk를 제한시켜 코너 궤적을 완화시켜주며, 기계부품의 생명을 연장시켜준다. 이는 가속도가 0에서 최고점으로 또는 그 반대로의 천이시간(I & III 구간) 최고 가속도의 지속시간(II 구간)

간의 비를 적절히 결정함으로써 얻을 수 있다. 이 지속시간의 비가 작으면 모션제어 패턴이 유연해지는 반면 동작에 소요되는 시간이 증가되게 된다. 즉, I 구간 동안 S-커브 프로파일은 짧은 거리를 이동하는 동안 저속으로 인하여 전체 동작에 소요되는 시간은 증가하게 된다. 반대로 지속시간의 비가 커져서 사다리꼴 프로파일처럼 II 구간과 같이 최고가속도를 주어진다 해도 매끄럽지 않고 원하지 않는 유해한 진동을 초래하여 실제로 동작 소요시간이 증가하게 된다. 이 같은 비는 정수 비율이 아니며 그에 따라 모든 연산이 비 정수 연산이 된다. 따라서 저가형태의 AVR 타입과 ARM 보드 타입에서 연산에 더 큰 영향을 초래하게 되며, 급속한 변화가 많은 서보모터에서 낮은 정확도와 연산 처리에 따른 지연시간 더욱 증가되어 시스템 불안정성에 기여하게 된다.

2. MCU 실수연산처리

마이크로프로세서의 경우 정수연산은 빠르게 처리할 수 있지만, 실수연산의 경우는 간단하지가 않다. 실수연산기능이 없는 많은 CPU는 수치보조연산기를 추가로 포함시켜야 하며, 웨이퍼면적과 전력 그리고 비용도 비싸게 된다. 따라서 저가 형태로 모션제어를 통한 실수연산처리를 하려면 비용이 증가되거나 충분한 안정성과 신속한 처리시간이 보장되지 않는다.

실수연산 기능이 없는 ARM 계열에서 간단한 실수와 정수 연산속도를 비교하는 실험에서 정수 덧셈과 곱셈은 루프를 1천만번 돌렸으며, 실수는 1백만번 돌려 2회 수행에 따른 시간차로 걸리는 시간을 측정하였다. 표 1에서와 같이 덧셈의 경우 1천만번 돌리는 데 걸리는 시간은 1천만 / 10 tick 으로 수행시간은 60 MIPS가 걸렸으며, 곱셈의 경우는 37 tick으로 총 연산처리 시간이 16.2 MIPS가 걸려 3.7배의 차이를 보였다. 실수의 경우 1백만번 돌려 걸린시간이 2.72 MFLOS와 1.2 MFLOS로 2.3배를 나타냈다. 정수와 실수의 처리시간을 비교하면 1초당 처리하는 횟수는 덧셈은 22.5 배, 곱셈은 13.5배정도 차이를 보였다. 이는 복잡한 연산 및 행렬계산에 있어서 제어기의 연산구조의 영향이 상당하다는 것을 짐작케 해준다. 기존의 제어기 설계에서 시간에 따른 가속시간과 감속시간을 프로파일을 통해 위치오차에 의한 제어기에 할당함으로써 목표위치에 빠르고 정확하게 수렴토록 설정하게 되는데, 실수연산과 목표위치를 빠르게 적용하지 못하는 제약이 생기게 된다.

표 1. ARM9 CPU에서의 정수/실수 연산처리 시간비교.

Table 1. Comparison of calculation time between integer arithmetic operation and floating-point one in ARM9 CPU.

	덧셈	곱셈
정수연산(1회)	53tick	98tick
정수연산(2회)	63tick	135tick
두 루프 시간차이	60MIPS	16.2MIPS
실수연산(1회)	59tick	53tick
실수연산(2회)	81tick	103tick
두 루프 시간차이	2.72MFLOS	1.2MFLOS

※ 정수 루프횟수(10,000,000), 실수 루프횟수(1,000,000)

- system clock = 67MHz
- 1 tick = 1/60sec(60Hz)
- 1MIPS = 1초당 실행되는 정수 연산의 백만단위
- 1MFLOPS = 1초당 실행되는 소수점 연산의 백만단위

III. S-커브 구현을 위한 감마보정 프로파일 기법

1. S-커브곡선 추정기법

일반적으로 BLDC 모터, 서보모터 등에 쓰이는 16bit 마이크로프로세서는 속도 프로파일 계산과 제어기능을 병행해야 하는데, 연산능력의 한계로 인해 속도 프로파일의 용도는 전적으로 시간에 따른 가감속구간에서의 추정값을 결정하여 제어에 할당하는 역할로 쓰이게 된다. 이러한 경우, 초기에 속도 프로파일의 정확성과 유연성, 처리시간이 중요하게 된다. 본 논문에서는 제안된 기법을 통해 가감속구간의 초기 속도 프로파일을 작성함으로써 정확성과 처리시간을 절감하여 기존방법에 의한 제약을 해결할 수 있었다.

제안된 방법은 주어진 최종 위치에 도달하기 위해 추정된 사다리꼴 프로파일을 S-커브로 만들어 모터에 입력시킨다. 이때 가속구간과 감속구간을 그림 4처럼 영상처리에 사용되는 감마보정 방법을 적용시켜 가속, 감속구간, 변곡점에서의 속도(deg/s)에 대한 jerk-bounded 궤적을 완화시켰다.

그림 4에서 가속구간을 지나 입력속도인 V_{max} 에 도달하는 시간이 T_2 일 때, 감마보정을 위한 직선의 중간점을 결정해야 된다. 이때 미리 알고 있는 V_{max} 와 T_2 의 절반 값을 감마보정의 중간값 V_h 로 결정한다. I 구간에서의 감마보정에 대해 식 (1) 과 같이 정의하여 V_h 속도 파라미터 이득을 얻는다. T_s 는 입력되는 시간, V_h 는 T_1 에서의 출력 속도, γ 는 감마보정계수이다. II 구간에서는 감마보정계수를 역 치환하여 식 (2) 와 같이 정리하였다.

$$V = \frac{V_{max}}{2} \times \left(\frac{T_s}{T_1}\right)^\gamma, \text{ for } 0 \leq T_s \leq T_1 \quad (1)$$

$$V = V_h + (V_{max} - V_h) \left(\frac{T_s - T_1}{T_2 - T_1}\right)^{1/\gamma}, \text{ for } T_1 \leq T_s \leq T_2 \quad (2)$$

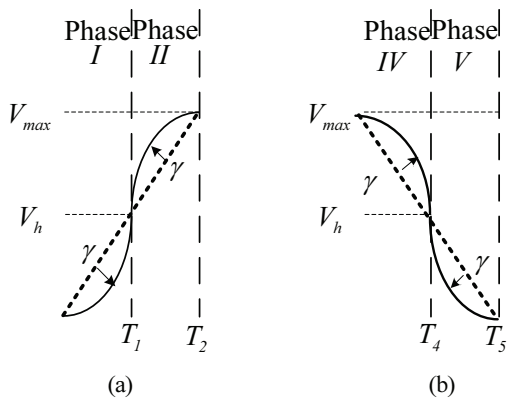


그림 4. 가속구간(a), 감속구간(b)에서의 감마보정 S-커브.
Fig. 4. Gamma corrected S-curve in (a) accelerated and (b) decelerated phases.

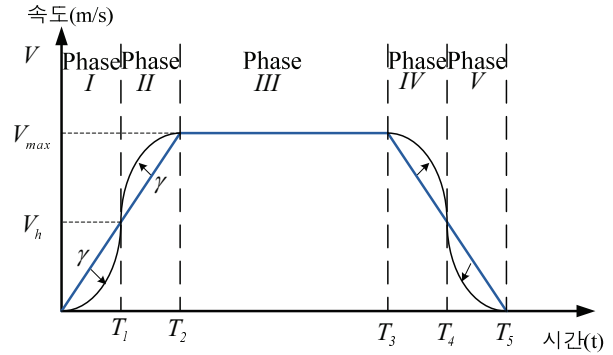


그림 5. 감마보정된 S-커브화 속도프로파일.
Fig. 5. S-curved velocity profile with gamma correction.

가속구간에서의 감마보정 적용식은 위와 동일하나 IV 구간동안은 식 (2)가 V 구간동안은 식 (1)이 적용된다.

가속구간과 감속구간에서의 감마보정 S자 곡선을 적용시킨 속도 프로파일을 그림 4와 같이 나타냈다. 그림 4에서 보듯이 가속구간과 감속구간에서 최고속도 V_{max} 의 중간점 속도(V_h)를 감마보정식에 적용하기 위해 두 개의 구역으로 나누어 가속도 증가 구간(I, IV 구간)과 가속도 감소 구간(II, V 구간)로 분리시킬 수 있다.

2. 감마보정 프로파일 작성

본 논문에서 제안한 감마보정 프로파일 작성을 위해 임의의 이동 각도에 대한 프로파일을 작성하여 그림 6과 같이 결과를 검증하였다. 위 조건의 프로파일을 작성하기 위해 최종 이동 각도는 153.134679°, 이동속도는 51.78945 deg/s 로, 가속, 감속크기는 1000 ms로 1 sec간 가속 및 감속을 하도록 임의의 파라미터를 구성하였다.

주어진 최종 이동 각도를 가속, 감속구간과 최고속도 구간으로 나누어 가속, 감속구간에서의 이동 각도는 각각 26.394725° 이며, 최고속도 구간에서는 100.345229°를 나타냈다. 이 때, 제어주기가 10 ms 이므로 1초 동안에 주어진 목표속도 즉 최고속도에 도달하는 것을 알 수 있다. 또한 감마보정과 중간 속도점을 통해 가감속 구간에서의 S-커브 유형이 나타남을 그림 7에서와 같이 나타냈다. 이는 위에서 언급한 감마 보정 커브 유형식에 대한 검증이 가능함을 나타낸다.

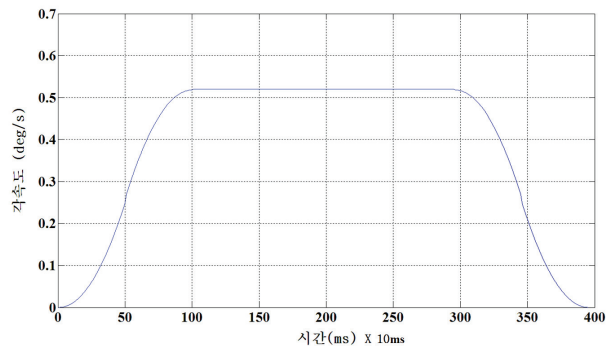


그림 6. 임의의 이동 각도에 대한 프로파일 결과.
Fig. 6. Resulting velocity profile for the arbitrary motion angle.

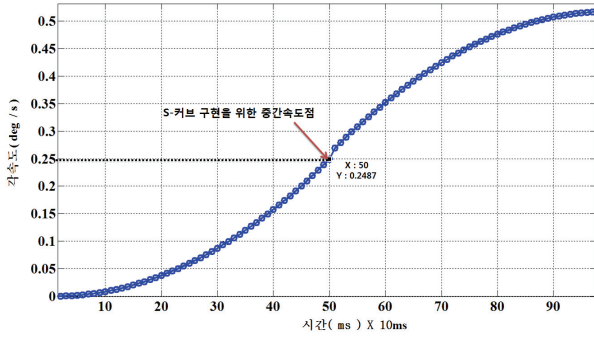


그림 7. 중간 속도점을 통한 감마보정 커브구현.
Fig. 7. S-curve by gamma correction about the median point.

IV. 빠른 모션제어를 위한 실수 연산처리 기법

1. 실수 연산 처리 기법

마이크로프로세서를 이용하여 모션제어를 하는 경우, 지연이 없는 연산처리 뿐만 아니라 목표위치에 빠르고 정확하게 위치이동을 시키는 프로파일 작성이 중요하다. 대부분은 비용절감을 이유로 소프트웨어적으로 실수연산을 처리하나, 이 또한 충분한 연산력 확보를 위한 웨이퍼면적의 증가나 추가적인 전력이 소요된다. 이 같은 추가적인 연산비용을 줄이기 위한 최적의 연산조건은 행렬의 반복연산을 줄이며, 실수 값을 처리하는데 있다. 본 논문에서 제안하는 기법은 제어기의 속도 프로파일을 추정함에 있어서 실수연산을 보장하고, 절제된 연산으로 프로파일을 생성하는 것이다.

기존의 S-커브 프로파일의 장점을 포함하여, 초기 저속에 따른 전체 전송시간의 증가를 줄여주고, 연산처리속도가 빠르고 실수처리가 가능함에 따라 정확한 목표 제어값 피드백 및 시간을 단축시킬 수 있다. 식 (3)-(6)은 제어주기에 따른 감마보정 속도 프로파일을 보여준다.

$$T_5 = T_2 + \frac{(d \times T_s)}{V_{max}} - 1 \tag{3}$$

$$T_{32} = T_5 - (2 \times T_2) = T_3 - T_2 \tag{4}$$

$$T_1 = \frac{T_2}{2}, \quad V_h = \frac{V_{max}}{2} = V_{max} - V_h \tag{5}$$

$$\delta = \frac{d \times T_s - V_{max} \times (T_2 + T_{32})}{T_{32}} \tag{6}$$

T_s 는 제어주기(10 ms), d 는 이동위치를 나타낸다. δ 는 소수점자리 오프셋 값을 나타내며, 실수연산이 가능하도록 조정하는 구간이다. T_{32} 는 최고속도의 시간크기이다.

본 논문에서 제안된 프로파일은 감마보정에 따른 속도프로파일의 초기 연산시간이 짧아 구간을 세 개로 나누어 처리하게 된다. 우선 속도 프로파일의 가속구간에서는 $t \geq 0$ 그리고 $t \leq T_2 - 1$ 으로 고려하여 원하는 속도 입력 값의 절반인 V_h 지점을 기준으로 시간 T_1 이 결정되며, 식 (7)의 $0 \leq t \leq T_1$ 에서 $V(t)$ 를 결정한다. 또한 커브곡선을 결정하기 위한 변수 γ 의 값에 의해 I과 II의 구간이 결정되게 된다. 따라서 $T_1 \leq t \leq T_2$ 는 II 구간에서의 속도 $V(t)$

를 나타낸다.

$$V(t) = \begin{cases} V_h \times \left(\frac{t}{T_1}\right)^\gamma, & \text{for } 0 \leq t \leq T_1 \\ V_h \left[1 + \left(\frac{t - (T_{21} - 1)}{T_{21}}\right)^{1/\gamma}\right], & \text{for } T_1 \leq t \leq T_2 \end{cases} \tag{7}$$

두 번째 구간은 최고속도구간으로 기존의 V_{max} 구간에 서 소수점자리 δ 값만큼 offset 시킨 구간으로 $V_{max+\delta}$ 는 식 (8)과 같이 나타내어 기존의 소수점까지도 속도 연산에 포함시켜 정확한 데이터처리를 가능하도록 한다.

$$V_{max+\delta} = V_{max} + \delta \tag{8}$$

$$V(t) = \begin{cases} V_{max} - V_h \left(\frac{t - T_{43} + 1}{T_{43}}\right)^{1/\gamma}, & \text{for } T_3 \leq t \leq T_4 \\ V_h \left(\frac{t - T_4}{T_{54}}\right)^\gamma, & \text{for } T_4 \leq t \leq T_5 \end{cases} \tag{9}$$

세 번째 구간은 감속구간으로 가속구간과는 반대로 $V(t)$ 를 식 (9)와 같이 정의한다. $T_{43} = T_3 - T_3$ 과 $T_{54} = T_5 - T_4$ 는 감속구간의 구간 IV와 구간 V의 시간크기이다.

따라서 감마보정식과 실수연산을 위한 오프셋 양의 값을 적용시켜 그림 8과 같이 구간 III의 기준에 적용되는 최고속도 V_{max} 에서 소수점 자리값 δ 을 증가시켜 설정된 최고속도 $V_{max+\delta}$ 를 연산할 수 있게 된다. 이를 임의의 각도에 대한 프로파일 실험에 적용하여 소수점 실수연산이 가능함을 그림 8과 같이 확인하였다. 16 MHz의 처리속도를 가지는 저가의 마이크로프로세서에 대한 배열 연산처리를 빠르게 하기 위해 제어주기의 처리를 10 ms 기준으로 했을 때 전체주기를 고려한 시간 T_s , 주어진 최고속도 $V_{max+\delta}$ 를 고려함에도 불구하고 빠르게 연산과정이 진행되는 것을 확인하였다.

이에 대한 기존의 그림 9에서 보듯이 1000 ms에서 S자형 곡선 모서리 부분의 적절히 분배된 소수점 오프셋 값이 더해져 최고속도 $V_{max+\delta}$ 가 추종됨을 볼 수 있으며, 그림 10에서는 최고속도구간 시작, 끝 지점을 확대하여 제안된 프로파일을 통해 실수연산 및 정밀도 높도록 소수점값 적용이 가능하며 보다 효율적임을 보여주고 있다.

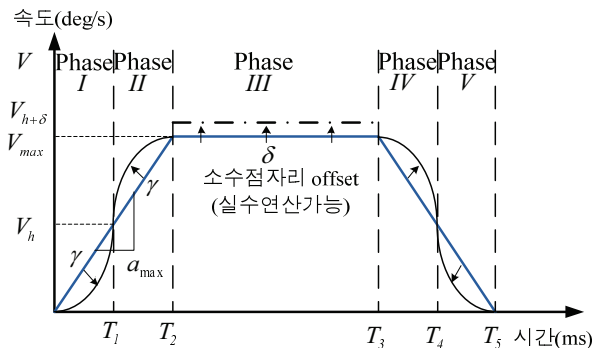


그림 8. 실수연산이 가능한 프로파일.
Fig. 8. Velocity profile capable of floating-point arithmetic.

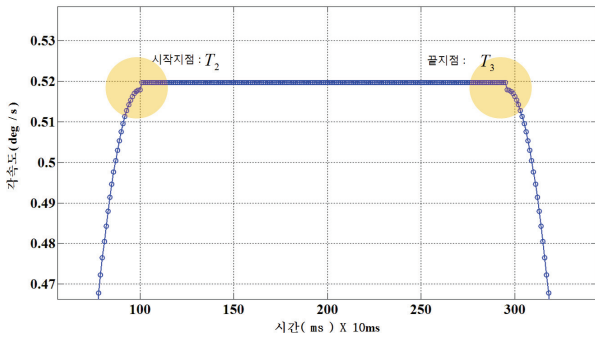


그림 9. 최고속도구간의 소수점 오프셋 값 적용 프로파일.
Fig. 9. Profile after applying offset value to the maximum velocity section.

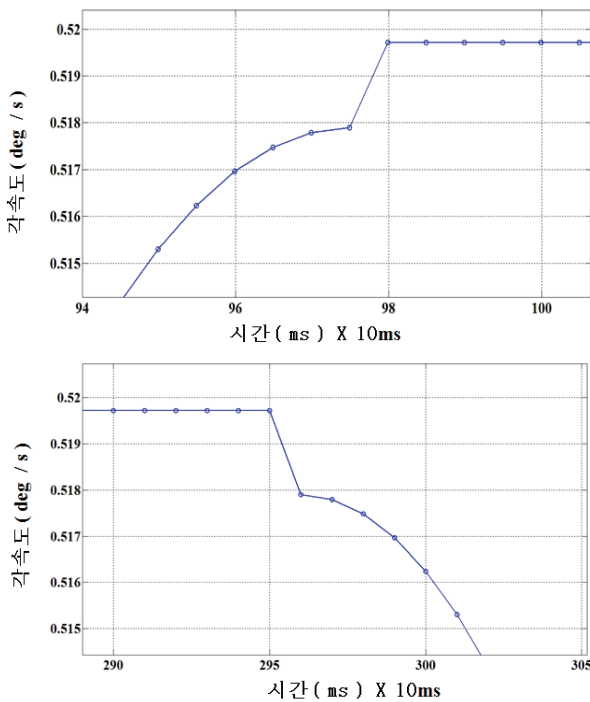


그림 10. 최고속도구간의 시작과 끝지점 확대 프로파일.
Fig. 10. Enlarged view of the starting and the ending point of the maximum velocity section.

링버퍼를 이용한 convolution 프로파일 생성방식이 가지는 초기처리속도 비교실험을 제 V 장 실험결과를 통해 확인하였다.

V. 실험결과

실험은 이미 기존에 널리 사용되는 링버퍼와 속도벡터의 실시간 convolution 방식과 본 논문에서 제안한 방식 2개를 비교하도록 한다. AVR사의 ATmega128의 16 MHz의 클럭을 가지는 마이크로프로세서로 제어주기 10 ms마다 이동해야할 속도를 계산하는데 걸리는 시간을 측정하였다.

먼저 기존 프로파일 생성 방식의 결과는 그림 11과 같이 제어주기에서 한번 속도가 계산되어지는데 약 750 μ s를 넘어서는 시간이 측정되었다. 그림 12에서는 본 논문에서 제

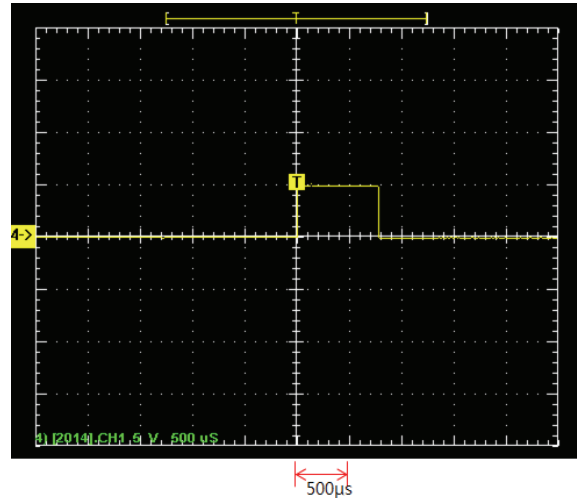


그림 11. 링버퍼 방식 프로파일 처리시간(500 μ s).
Fig. 11. Profile processing time(500 μ s) of the ring-buffer method.

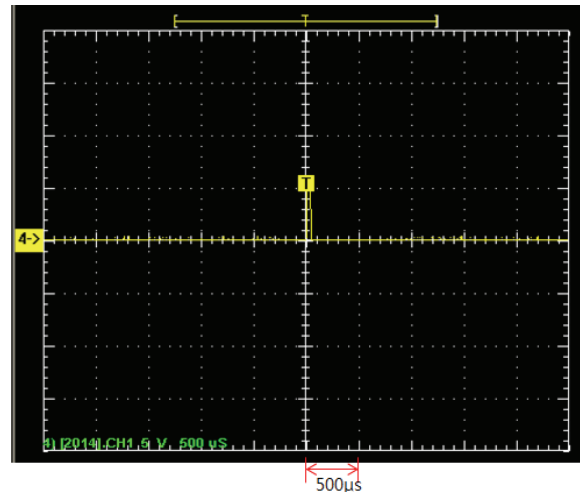


그림 12. 제안된 방식의 프로파일 처리시간(500 μ s).
Fig. 12. Profile processing time(500 μ s) of the proposed method.

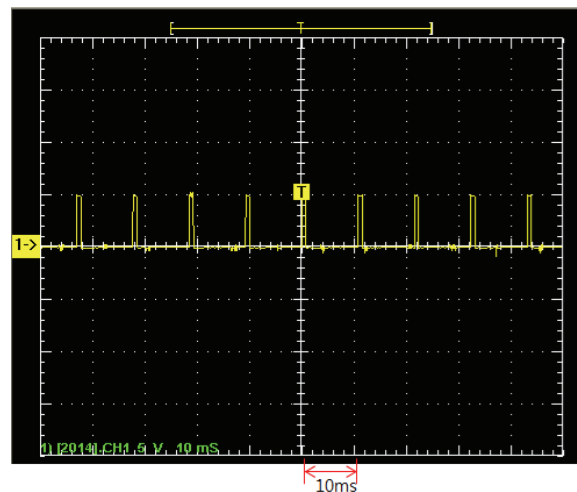


그림 13. 링버퍼 방식 프로파일 처리시간(10 ms).
Fig. 13. profile processing time(10 ms) of the ring-buffer method.

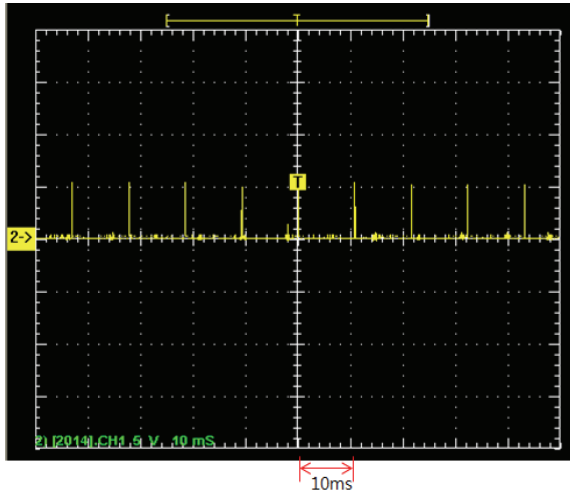


그림 14. 제안된 방식의 프로파일 처리시간(10 ms).

Fig. 14. Profile processing time(10 ms) of the proposed method.

안하는 프로파일 생성 방식을 이용하여 측정한 결과 제어 주기에서 속도를 이끌어내는데 약 $40\mu s$ 가 소비되었다. 결과는 이미 예상한 바와 같이 기존 방식의 배열연산, 배열연산을 위한 반복문의 쓰임으로 인해 큰 차이를 보였다. 제안된 방식의 경우 제어 주기 시간만큼의 프로파일 크기 및 최고속도 및 오프셋 구간 크기만을 구하여 각 제어 시간에 대해 간단한 산술 연산을 하기 때문에 아주 큰 계산시간의 차이를 보였다. 실험 결과는 1개의 속도 프로파일을 대상으로 한 실험 결과이므로 여러 모터를 하나의 마이크로프로세서에서 사용한다고 가정했을 경우 실행시간은 점점 큰 격차를 보이게 될 것이다.

VI. 결론

본 논문에서는 저가의 범용 8비트 프로세서에서 쉽게 사용할 수 있도록 속도 프로파일 생성 지연 시간을 단축한 프로파일 생성 기법을 제안하였다. 제안된 방식의 기존의 사다리꼴 및 S-커브 형태의 프로파일을 모두 지원하도록 설계되었고, 특히 정밀 이동에 적합하도록 프로파일 생성시 프로파일에 포함시키기 어려운 소수점 아래의 값을 오프셋 값으로 적절히 분배하여 구동 프로파일 시간에 분포시킴으로써 정밀도 높은 프로파일이 생성되도록 하였다. 이를 증명하기 위해 16 MHz의 처리 속도를 가지는 저가의 8비트 MCU를 이용하여 제어 주기 10 ms에서 링버퍼를 이용한 프로파일 생성 방식과 제안된 방식의 실행 속도를 비교하여, 프로파일 생성 지연이 없음을 증명하였다. 증명된 결과는 단 하나의 프로파일 생성을 비교하였으나 많은 산업현장 및 연구실 레벨에서 하나의 MCU에 대체로 많은 모터를 이용하므로 그 효율은 모터 사용 개수가 많을수록 더욱더 효율적 방식이 되리라 판단된다.

참고문헌

- [1] I. Boldea and S. A. Nasar, *Linear motion electromagnetic systems*, John Wiley & Sons, Inc., 1985.
- [2] Y. B. Cha, D. H. Song, B. M. Koo, M. Y. Park, J. A.

Kim, and J. K. Choi, "The design of an auto tuning pi controller using a parameter estimation method for the linear BLDC motor," *The Journal of the Korea Institute of Maritime Information & Communication Sciences*, vol. 10 no. 4, pp. 659-666, 2006.

- [3] M. H. Kwon, Y. S. Kang, C. H. Kim, and G. T. Park, "A path & velocity profile planning based on a algorithm for dynamic environment," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 5, pp. 405-411, 2011.
- [4] J. Y. Ahn, H. H. Cho, H. R. Cha, S. Y. Kang, and K. H. Kim, "A study on the determination of adequate velocity profile for position control of linear BLDCM," *Proc. of the Conference on Power Electronics, Korea Power Electronics Society*, pp. 501-503, Jun. 2006.
- [5] P. S. Jang, J. Y. Lee, J. H. Jung, and C. H. Hyun, "Torque control of DC motor using velocity profile based acceleration/deceleration control," *Proc. of the KIIS Fall Conference 2011*, vol. 21, no. 2, pp. 189-190, 2011.
- [6] H. D. Kim and H. R. Choi, "Design and analysis of an output feedback controller for a chain of integrators system compensating measurement noise of feedback sensor," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 17, no. 4, pp. 299-303, 2011.



문상찬

2010년 경희대 기계공학과 졸업. 2012년 동 대학 석사. 2012년~현재 동 대학 기계공학과 박사과정재학. 관심분야는 계측제어, 자동차 항법 & 위치제어.



김재준

2006년 동의대학교 메카트로닉스과 졸업. 2009년 경희대학교 기계공학과 석사. 2009년~2010년 동 대학 기계공학과 박사과정재학. 2012년~현재 한국생산기술연구원. 관심분야는 로봇제어 및 응용, 머신비전, 기계학습.



남규민

2009년 조선대 기계공학과 졸업. 2011년 경희대학교 기계공학과 석사. 2012년~현재 세종공업사 선형연구소. 관심분야는 Walking Robot, Motion Control.



김 병 수

1991년 서울과학기술대 기계설계과 졸업. 1993년 경희대학교 기계공학과 석사. 2011년 동 대학 박사. 2011년~현재 경희대학교 기계공학과 박사후연구원. 관심분야는 Mobile & Walking Robot.



이 순 걸

1983년 서울대 기계공학과 졸업. 1985년 KAIST 대학원 석사. 1993년 Univ. of Michigan 박사. 1994년~1996년 KIST 휴먼로봇센터 선임연구원. 1996년~현재 경희대학교 기계공학과 교수. 관심분야는 로봇 제어, 메카트로닉스, 인체공학.