



$$\phi^k(t+k|t) = \frac{F(q^{-1})}{Pd(q^{-1})C(q^{-1})}y(t) + \left[ \frac{E(q^{-1})B(q^{-1})}{C(q^{-1})} + Q(q^{-1}) \right] u(t) - Rq(t) \quad (10)$$

이 경우에 제어칙은 식(10)의  $\phi^k(t+k|t)$ 가 영이 되는  $u(t)$ 이다. 따라서 제어칙은 다음과 같이 구해진다.

$$u(t) = \frac{C(q^{-1})R(q^{-1})w(t) - F(q^{-1})y(t)/Pa(q^{-1})}{E(q^{-1})B(q^{-1}) + C(q^{-1})Q(q^{-1})} \quad (11)$$

$A(q^{-1}), B(q^{-1})$  및  $C(q^{-1})$ 가 미지이면 예측모델의 식(8)을 사용하여 자기동조 제어기의 제어칙을 구성해야 한다.

식(5)에서 시점  $t$ 의 신호  $Q(q^{-1})u(t-k)$ 와  $R(q^{-1})w(t-k)$ 는 알 수 있는 값이므로  $\phi(t)$ 를 예측하는 문제는 단순히

$$\phi y(t) = P(q^{-1})y(t) \quad (12)$$

을 예측하는 문제로 된다. 식(1)과 식(7)에서

$$\phi y(t+k) = \frac{F(q^{-1})}{Pd(q^{-1})C(q^{-1})}y(t) + \frac{E(q^{-1})B(q^{-1})}{C(q^{-1})}u(t) + E_f^k(t+k) \quad (13)$$

이때 필터된 출력  $y_f(t)$ 를 다음과 같이 정의한다.

$$y_f(t) = \frac{y(t)}{Pa(q^{-1})} \quad (14)$$

이 경우  $G(q^{-1}) = E(q^{-1})B(q^{-1})$ 이라 하고  $C(q^{-1}) = 1$ 이면 식(13)는 다음과 같이 된다.

$$\phi y(t+k) = F(q^{-1})y_f(t) + G(q^{-1})u(t) + \epsilon(t+k) \quad (15)$$

여기서  $\epsilon(t+k)$ 는 잡음  $\xi(t)$ 의 이동평균이다.  $F(q^{-1})$ 와  $G(q^{-1})$ 의 추정치는 최소자승법을 사용하여 얻을 수 있다.  $k$ 단 예측치는

$$\begin{aligned} \phi y(t) &= X^T(t-k)\hat{\Theta}(t) \\ X^T(t) &= [ y_f(t-k), y_f(t-k-1), y_f(t-k-2), \\ &\quad ; u(t-k), u(t-k-1) ] \\ \hat{\Theta}^T(t) &= [ f_0(t), f_1(t), f_2(t); g_0(t), g_1(t) ] \end{aligned} \quad (16)$$

미지의 매개변수  $\Theta$ 는 다음과 같이 축차형태로 추정될 수 있다.

$$\hat{\Theta}(t) = \hat{\Theta}(t-1) + P(t)X(t-k)[y_f(t) - \phi y(t-1)] \quad (17)$$

$$P(t) = \frac{1}{\mu} [ P(t-1) - \frac{P(t-1)X(t-k)X^T(t-k)P(t-1)}{\mu + X^T(t-k)P(t-1)X(t-k)} ] \quad (18)$$

$F(q^{-1}), G(q^{-1})$  파라미터의 수를 각각  $L, J$ 라고 하면 다음과 같이 된다.

$$L = \deg A(q^{-1}) + \deg Pd(q^{-1}) \quad (19)$$

$$J = \deg B(q^{-1}) + \deg E(q^{-1}) + 1 = \deg B(q^{-1}) + k \quad (20)$$

$G(q^{-1}) = E(q^{-1})B(q^{-1}), C = 1$  그리고  $F(q^{-1}), G(q^{-1})$ 을 식(11)에 대입하면 식(21)이 구해진다.

$$u(t) = \frac{R(q^{-1})w(t) - F(q^{-1})y(t)/Pa(q^{-1})}{G(q^{-1}) + Q(q^{-1})} \quad (21)$$

### 2.2 PID구조를 갖는 자기동조 제어기

여기서는 제어하고자 하는 이산시스템을 설정한 후 이를 제어하기 위해 연속시간형 PID제어기 구조로부터 이산시간형 PID구조를 유도한다. 표준형 연속시간 PID 제어기는 다음과 같이 정의 된다.

$$U(s) = Kp \left( 1 + \frac{1}{Tis} + Td \cdot s \right) E(s) \quad (22)$$

$$E(s) = W(s) - Y(s) \quad (23)$$

$U(s), E(s), W(s)$  및  $Y(s)$ 는 각각 입력, 오차, 목표치 및 출력의 라플라스 변환이고,  $Kp$ 는 비례이득,  $Ti$ 는 적분(동작)시간,  $Td$ 는 미분동작시간이다. 그림1은 일반적인 PID제어기의 구조를 나타내고 있다.

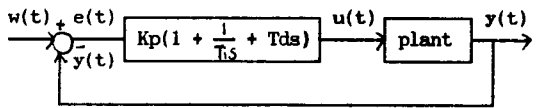


그림1 일반적인 PID제어기의 구조  
Fig.1 Structure of a general PID controller

식(21)을 이산시간영역에서 표현하면 다음과 같다.

$$u(t) = Kp \left[ 1 + \frac{1}{Ti} \frac{T_s}{1-q^{-1}} + \frac{Td}{T_s} (1-q^{-1}) \right] e(t) \quad (24)$$

여기서  $T_s$ 는 샘플구간이다. 식(24)의 제어입력은 플랜트에 가해지기 때문에 위치형(position form)이라고 한다. 제어입력의 1단변화분은  $\Delta u(t) = u(t) - u(t-1)$ 이므로 속도형(velocity form)이라고 한다. 이 때(24)식은 다음과 같이 된다.

$$\Delta u(t) = Kp \left[ 1 - q^{-1} + \frac{T_s}{Ti} + \frac{Td}{T_s} (1 - 2q^{-1} + q^{-2}) \right] e(t) \quad (25)$$

$$\text{여기서 } e(t) = w(t) - y(t) \quad (26)$$

목표치는 적분항에만 나타나고,  $y_f(t)$ 는  $y(t)$ 를 필터링한 것이라면 속도형 PID제어기는 다음과 같이 쓸 수 있다.

$$\begin{aligned} \Delta u(t) &= Kp \left[ e(t) - e(t-1) + \frac{T_s}{Ti} e(t) \right. \\ &\quad \left. + \frac{Td}{T_s} [ -y_f(t) + 2y_f(t-1) - y_f(t-2) ] \right] \end{aligned} \quad (27)$$

식(27)은 다음과 같이 정리 될 수 있다.

$$\begin{aligned} \Delta u(t) &= Kp \left[ \left( 1 + \frac{T_s}{Ti} - q^{-1} \right) w(t) - \left[ \left( 1 + \frac{T_s}{Ti} + \frac{Td}{T_s} \right) \right. \right. \\ &\quad \left. \left. - \left( 1 + 2\frac{Td}{T_s} \right) q^{-1} + \frac{Td}{T_s} q^{-2} \right] y_f(t) \right] \end{aligned} \quad (28)$$

자기동조제어기가 PID구조를 갖기 위해서는 제어기는 적분동작을 해야하고 다항식  $F(q^{-1})$ 의 차수는 2이어야 한다. 식(19)에서 다항식  $F(q^{-1})$ 의 차수가 2라는 것은  $L=3$ 과 등가이다. 적절한 출력필터의 차수는  $\text{deg}P_d(q^{-1})=1$  또는  $\text{deg}P_d(q^{-1})=0$ 이다. 그러면 식(19)로부터  $L=3$ 이라는 것은 2차나 3차의 프로세스모델을 선택하는 것과 등가임을 알 수 있다. 즉  $n=2$  또는  $n=3$ 이다. 따라서

$$P_a(q^{-1}) = \frac{1+q^{-1}P_{a1}}{1+P_{a1}} \quad (29)$$

로 가정한다. 다음  $R(q^{-1})$ 는 설계시 임의로 지정할 수 있으므로

$$R(q^{-1})w(t) = [f_0(t) - f_2(t) + (f_1(t) + 2f_2(t))q^{-1}]w(t) \quad (30)$$

로 둔다.  $Q(q^{-1})$ 도 마찬가지로 설계자가 임의로 지정할 수 있으므로 자기동조제어식에 적분동작이 들어가지도록 다음과 같이 선택한다.

$$G(q^{-1}) + Q(q^{-1}) = \frac{1-q^{-1}}{v} \quad (31)$$

여기서  $v$ 는 자기동조제어기를 동조시키는데 편리한 수단을 제공해 주는 설계파라미터이다. 식(31)과 (30)을 식(21)에 대입하면 다음의 PID구조를 갖는 자기동조제어기가 얻어진다.

$$u(t) = v[(f_0(t)-f_2(t)) + (f_1(t)+2f_2(t))q^{-1}]w(t) - [f_0(t)+f_1(t)q^{-1}+f_2(t)q^{-2}]y_f(t) \quad (32)$$

PID제어기와 대응되는 식은 식(28)과 식(32)을 서로 비교하여 얻어진다.

$$K_p = -v(f_1+2f_2) \quad (33)$$

$$T_d = \frac{vf_2T_s}{K_p} \quad (34)$$

$$T_i = \frac{K_p T_s}{v(f_0+f_1+f_2)} \quad (35)$$

식(30)과 (31)을 만족하도록 시변의  $Q(q^{-1})$ 와  $R(q^{-1})$ 을 갖는 예측모델 식(15)를 사용하는데  $Q(q^{-1}), R(q^{-1})$ 를 명확하게 계산할 필요는 없다. 왜냐하면  $\phi(t)$ 에서와는 달리  $\phi y(t)$ 에 있어서는  $Q(q^{-1})$  및  $R(q^{-1})$ 이 특별히 정의되지 않았기 때문이다. 식(35)에서  $v$ 는 총래의 제어기 이득  $K_p$ 와 같은 효력을 갖는다. 따라서  $v$ 의 큰 값은 부족제동응답, 작으면 과제동응답의 결과로 얻어진다.

### 3. 하드웨어 설계 및 실험

본 연구에서는 계산속도를 높이기 위하여 8088 CPU 2개를 가지고 다중 processor 제어기를 설계하여 계산량을 분배하였다. 또한 연산속도를 높이기 위하여 8087 (coprocessor)도 사용하였다.

#### 3.1 병렬통신회로

그림2와 같이 2개의 processor간의 병렬통신에는 8255(PPI)와 LS123의 저항값과 커패시터값을 조절하여 사용하였다.

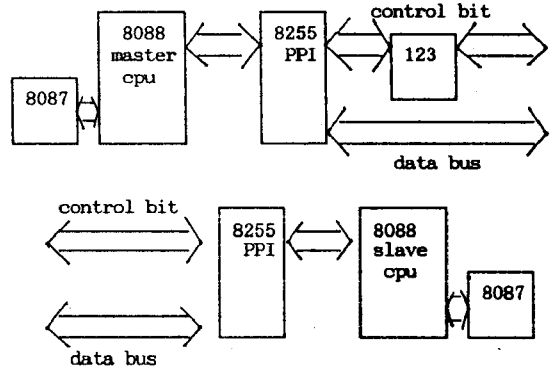


그림2 병렬통신회로  
Fig.2 Parallel communication circuit.

### 3.2 전동기 제어시스템

전동기 속도는 전동기에 부착된 encoder로 측정되며 LS123을 이용한 단위시간당 펄스의 밀도로 제어입력을 조절하였다.

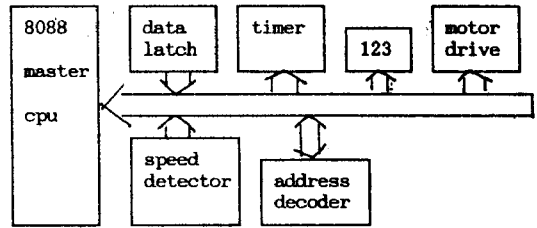
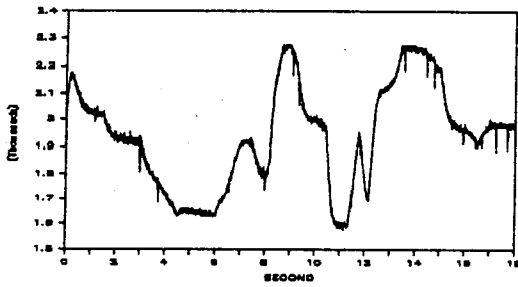


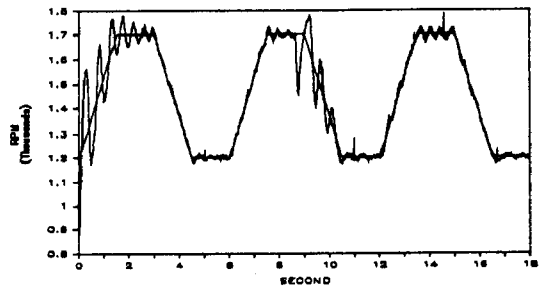
그림3 전동기 제어시스템  
Fig.3 Motor control system

### 4. 실험결과

본 연구에서는 고전적인 PID알고리즘과 자기동조 PID 알고리즘을 DC서보전동기 속도제어에 적용하였다. 또한 주어진 서보전동기에 걸정적인 부하변동이 생겼을 때의 각 알고리즘의 부하회복 상태를 모의실험을 통해 확인하고 실험을 통하여 입증하였다. 기준 입력은 사다리꼴 입력을 사용하였고 고전PID의 이득은 부하변동이 없을 때의 최적항을 선정하였다. 그림4는 부하가 감소할 수 있는 실험장치를 사용하여 9초에서부터 부하를 가했을 경우의 PID제어방식에 대한 응답을 보여준다. 그림5는 PID경우와 같은 부하조건에서의 자기동조 PID제어 방식에 대한 결과이다. 아래 그림에서 알 수 있듯이 초기 상태에서는 고전적인 PID제어기가 자기동조 PID제어기보다 상당히 빨리 기준 입력에 수렴해감을 볼 수 있지만 부하변동에 의해 모델의 매개변수가 영향을 받았을 경우, 고전PID제어기보다 자기동조 PID제어기가 부하변동에 따른 회복속도가 우수함을 보이고 있다.



(a)



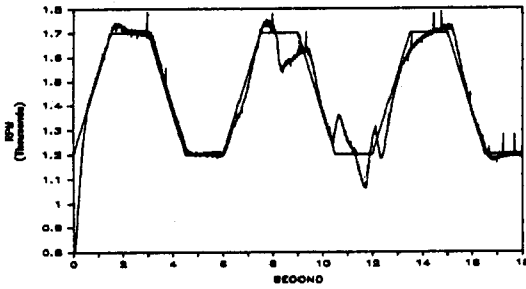
(b)

그림 5. 자기동조 PID 제어 방식

a) 입력 b) 출력

Fig.5 Self-tuning PID control method

a) input b) output



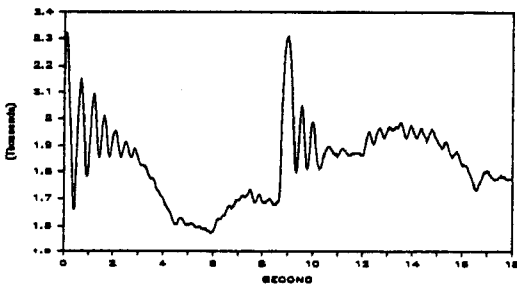
(b)

그림 4. PID 제어 방식

a) 입력 b) 출력

Fig.4 PID control method

a) input b) output



(a)

### 5. 결론

본 연구는 기존의 PID제어기와 같은 간단한 구조를 가지면서 자기동조 제어기의 특성을 그대로 유지하는 자기동조PID 알고리즘을 DC서보 전동기 속도제어에 적용하였다. 제어기의 하드웨어 설계시에는 다중processor를 사용하여 1개의 processor를 사용하는 기존의 제어기 보다 계산속도를 높였으며, 따라서 계산량이 많은 적응제어 알고리즘인 자기동조PID 알고리즘을 무리없이 구현할 수 있었다. 그리고 부하가 가해졌을 경우 자기동조PID제어기가 매개변수를 추정하지 않는 고전적인 PID제어방식보다 수렴속도 및 부하에 대한 회복능력이 있어서 월간 우수함을 실제 실험을 통해 입증하였다.

다중processor 제어기 설계시 중요한 사항은 통신시간을 줄이는 것이다. 16bit, 32bit의 processor 및 통신능력이 강화된 IC를 채택하면 제어기의 부품수를 줄여 신뢰성을 높이는 한편 시스템의 스피드와 효율을 극대화 할 수 있을 것이다.

### 6. 참고문헌

1. P.J. Gawthrop, "Self-tuning PID controllers; Algorithm and Implementation," IEEE Trans. Automat. Contr., Vol.AC-31, No.3, pp.201-209, 1986
2. F.Cameron, D.E Seborg, "A self-tuning controller with a PID structure," Int.J.Cont., Vol.38, No. 32, pp.401-417, 1983.
3. B.Witenmark, "Self-tuning PID-Contrllers Based on Pole-placement," Dept. of Auto. Contr., Lund Inst. of Tech., Lund, Sweden, Rep., No. LUF2/(TFRT-7179/0371/, 1979.
4. D.W.Clarke, "Implementation of self-tuning controllers", in C.J. Harris and S.A. Billings (Eds) Self-tuning and adaptive control. Peter Peregrinus. 1981.
5. 양혜원, 강항수, "극베치 PID자기동조 제어방식에 의한 DC서보전동기 속도제어에 관한 연구." Trans. KIEE, Vol 37. No.9 SEP. 1988
6. B.C.Kuo, Incremental motion control, SRL Publishing company 1978.
7. Ray Duncan, Advanced MS-DOS, Microsoft 1986.