

공정 제어에의 應用을 위한 마이크로 컴퓨터 시스템의 개발*

李泰鎬 · 朴源深 · 禹治水 · 李秀東 · 黃德浩**

전 기 공 학 과

〈요 약〉

마이크로 프로세서를 이용한 디지털 공정제어 수단이 고찰되었다. 관심의 초점은 분산-종합제어에 적합한 단말제어 프로세서에 두었다. 시험 제작된 프로세서는 8개의 제어 루우프를 운전하는 독립된 제어장치이며 동시에 종합제어를 위하여 상위 전산기와 데이터 교환이 가능하도록 설계 되었다.

주기적 샘플링에 의한 비례-적분-미분(PID)제어방식을 기본으로 하고 있으며 파라메터 값에 의하여 PI, PD등으로 축소될 수 있도록 하였다. 시험 장치에서는 주기를 2초로 하였는데 A/D변환기의 적절한 선정에 의하여 0.2초정도까지는 쉽게 단축할 수 있다. 조작자용의 프로세서에 대한 고찰은 최소한의 기능을 가진 것에 한정하였으며 이는 실제 저용 현장과 관련하여 기능이 확장되어야 할 것이다.

사용된 마이크로 프로세서는 범용의 것으로 단말용에는 M6800계통을, 조작자용에는 SC/MP계통을 사용하였다.

Development of Microcomputer System for the Application to Process Control

Tai Ho Lee, Won Shim Park, Chi Soo Woo, Soo Dong Lee
and Duk Ho Hwang
Dept. of Electrical Eng.

〈Abstract〉

A microprocessor based digital process controller has been studied. The objective was to develop a stand-alone terminal processor for distributed on-line control systems.

The prototype terminal proposed in this paper has the capability of handling eight independent control loops under the supervisory control of main computer and/or operator console.

The control algorithm follows analog PID action based on periodically sampled data. The control action can also be reduced to PI, PD and so forth through parameter assignment. Availability of A/D converters to the authors limited sampling period to 2 seconds, but it could easily be reduced to 0.2 sec. employing appropriate A/D devices.

The number of operational functions of console processor is kept minimum in this study, the expansion of which should be done for particular application. Hardware implementation was done with general purpose microprocessors; M6800 system for terminal and SC/MP system for console.

I. 서 론

1960년 영국의 ICI사에 의하여 디지털방식에 의

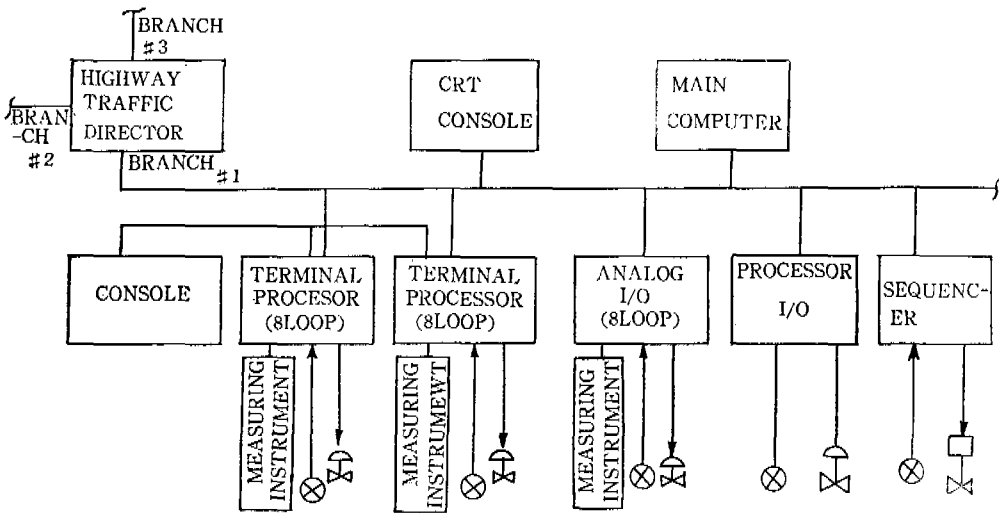
한 공정제어의 시도가 최초로 이루어진 이래 DDC (Direct Digital Control)의 연구도 끊임없이 진행 되어 왔으나 (1)(2)(3)(4), 그 유리 한과 아울러 여

*본 研究는 文教部 정책과제 연구비의 지원으로 이루어졌음.
*韓國電力(株)

러가지 문제점을 안고 있어서 일반 산업제어의 보급이 어려웠다. 즉 단일 제어장치에 의한 집중제어 방식은 제어대상과 제어장치와의 1대 1의 구체적 관련이라는 종래의 개념에서 근본적으로 벗어나야 하므로 기존 생산기구의 인적 또는 계통적인 대 개혁을 필요로 하고 단일 데이터 전송로 등의 일부에 고장이 생기면 전 시설의 운전이 정지된 위험을 내포하고 있다는 불안등이 파르프로 기존산업 조직으로부터 우선 심리적 저항을 받을 요소를 가지고 있다.

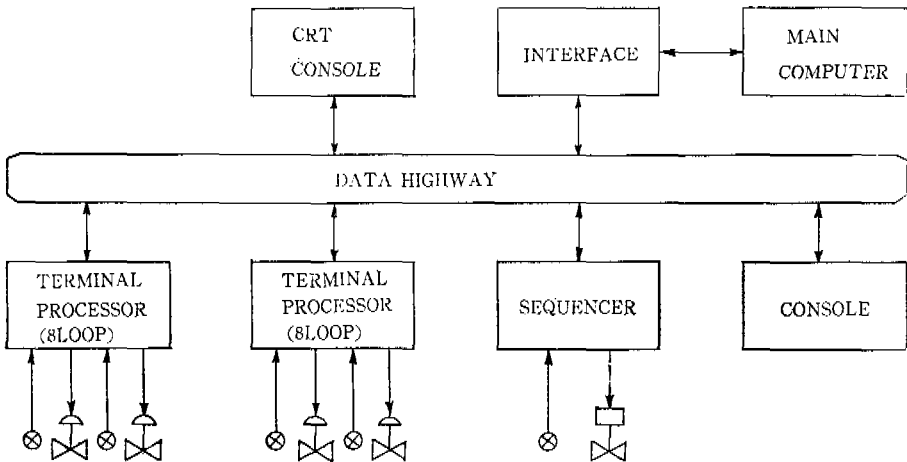
최근 각종 마이크로 프로세서들이 개발, 보급되어 위의 문제들 중 많은 부분을 해소할 수 있는 가

능성이 열리게 되었고 각국의 대 메이커들이 독자적 계통들을 소개하기에 이르렀다(4)(5)(6). 그림 1은 그 중 두 예를 보인 것인데 단말프로세서—조작자 프로세서—주전산기라는 3자의 유기적 관련운전을 모색하고 있다. 이 제어 기구의 최하층 구조를 이루고 있는 단말 프로세서는 제어 작용을 직접 담당한다는 점에서 종래의 공기식 또는 전자식 단일 루프 특정 제어장치와 같은 역할을 하면서 동시에 상위 전산기나 교신하여 종합제어를 가능하게 하는 양면의 중요성을 지니고 있다.



1-a. TDCS의 시스템구조

그림 1. DDC계통의 예



1-b. MICREX의 시스템구조

국내의 계측기 및 제어장치 산업은 극히 미약하며 산업체에 설치 운영되고 있는 공기식 또는 전자식 제어장치들이 거의 대부분 외국회사에 의존하고 있는 형편이니 전산기 제어에 이르러서는 연구조차 전무한 실정이다. 장래의 프래트들의 제어 방식이 전산화 되리라는 전망에 비추어 이 방향의 개발도 시급히 이루어져야 하리라고 생각된다.

본 연구에서는 마이크로 프로세서를 이용하여 범용 단말 프로세서를 설계, 제작하고 제어 프로그램을 개발하였다. 시작점은 8개의 제어 루프에 대한 제어요소로서의 기능을 가지며 동시에 상위전산기와 교신하여 필요한 데이터를 교환할 수 있는 표준형태를 모색하였다. 프로그램 개발에는 울산공대와 LOUGHBOROUGH대학이 공동 개발한 크로스 아셈블러를 이용하였으며 사용한 전산기는 CTL MODULAR ONE이다.

II. 단말 프로세서의 기능 설계

1. 기능 설정

단말 프로세서는 수개내지 수십개의 독립된 제어 루프를 수용할 수 있어야 하며 상위 기구의 계속적인 지령이나 도운없이 독자적으로 할당된 제어루프들의 동작을 유지하여야 한다. 또한 필요할 때는 조작자 및 상위 전산기와 교신하여 자료를 전송하거나 지령을 접수할 수 있어야 한다.

본 연구에서는 이러한 기능을 다음과 같이 설정하였다.

가. 단말 프로세서 1개가 담당할 제어 루프의 수는 8개로 한다. 16 또는 32개 루프로 수를 늘이는 것은 간단한 일이지만 그만큼 고장에 대한 영향도 커진다.

나. 제어의 방식은 PID를 표준으로 하며 P, I 및 D에 관련된 파라미터들의 값에 의하여 P, PI 등의 형태가 결정된다. 종속제어 등은 일단 고려하지 않는다.

다. 상위 기구와의 교신을 위한 데이터 하이웨이는 평상시에는 단말 장치로 부터 단절(OFF-LINE)되어 있도록 하며 필요에 의한 접속(ON-LINE)의 주도권은 단말 프로세서에 부여한다. 이 과정은 ① 인터럽트→② 승인→③ 하이웨이의 접속, 이라는 단계에 의하여 이루어지는데 인터럽트가 어느 편에서 발생되었는지에 관계없이 하이웨이 접속의 작업은

단말 장치가 담당한다. 이러한 방법은 전체 제어계통이 단순화되지만 융통성이 적어질 염려가 있으나 상위 기구의 인터럽트는 고장시의 긴급 점검이나 장치 교환 후 초기 운전에 한정되도록 단말프로세서의 동작 알고리즘이 설계되었으므로 문제가 없을 것이라고 생각된다.

라. 제어 방식과 파라미터 :서론에서 언급한 바와같이 제어 방식은 PID에 의존하기로 하였다. 단일 루프 제어 방식으로는 더 낮은 DDC고유의 수단이 개발되어 있지 않으며 샘플링 시간만 적당히 잡으면 아나로그 장치와 흡사한 결과를 얻을 수 있으므로 마이크로 프로세서 터미날에 적합한 방식이다(3)(7).

제어 연산의 아나로그 표현식은 다음과 같다.

$$m(t) = K_P e(t) + K_I \int e(t) dt + K_D \frac{de(t)}{dt} \quad (1)$$

$$e(t) = r(t) - b(t) \quad (2)$$

여기 사용된 기호들은 다음과 같다.

$m(t)$: 조작량

K_P : 비례감도

K_I : 적분제어계수 = 비례감도 × 적분계수

K_D : 미분제어계수 = 비례감도 × 미분계수

$e(t)$: 편차 또는 동작신호

$r(t)$: 기준치

$b(t)$: 제환량

식(1)의 연산을 위해서는 기준치, K_P, K_I 및 K_D 의 4개 파라미터가 결정되어야 하며 이들은 데이터 하이웨이를 통하여 상위기구로 부터 공급되어 진다. 제어 연산은 다음 절에서 설명하는 바와 같이 주기적으로 반복되며 매주기마다 한번씩 제어 파라미터들을 새로이 확인하게 되므로 필요한 때는 파라미터의 변환이 가능하다. 소규모의 계통에서는 단말 프로세서를 단독으로 제어 장치로 사용할 수 있으며 이때에는 파라미터를 포텐쇼메타에 의해서 조정하도록 하는 것이 편리하고 포텐쇼메타들에 기억 소자의 번지를 할당해 주기만 하면 된다.

2. 프로세서의 동작

단일 제어 루프의 동작의 관점에서 보면 본 연구의 단말 프로세서는 그림 2와 같이 단순한 PID 제어 요소에 불과하다. 이 제어 요소에 전절에서 설정된 기능을 부여하면 단말 프로세서의 동작 프로우 차트는 그림 3과 같이 되며 실제 프로그램을 부록에 실었다.

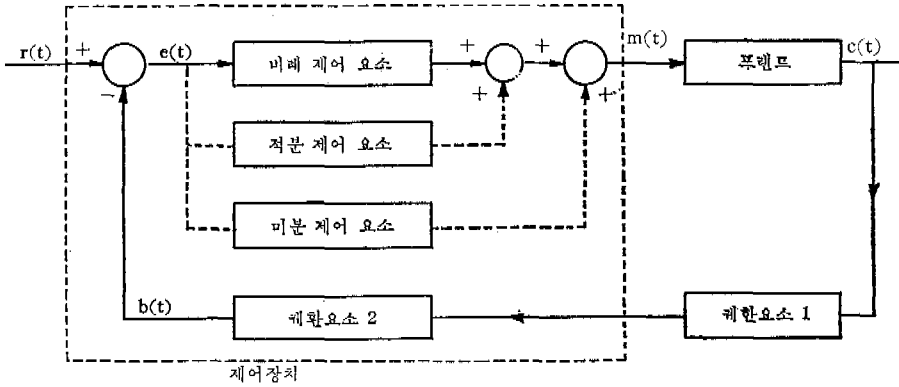


그림 2. PID제어계통도

처음 장치가 리셋되던 시스템 초기조정(Initialization)이 이루어진다. 이 단계에서 결정되어야 하는 것들로는 인터럽트 썬어비스 루우틴 포인터, 스택 포인터, PIA(Peripheral Interface Adapter) 상대 선정 등인데 본 연구에서는 EVALUATION-KIT를 사용하였으므로 NMI(Nonmaskable Interrupt) 썬어비스와 스택 포인터는 KIT에 포함된 JBUG 모니터 ROM의 기능을 이용하고 나머지 사항만 추가하였다[9][10].

초기 조정에 이어서 타이밍 신호를 받아들여 정상 운전 루우틴에 들어가게 되면 먼저 조작자측의 기억장치(MEMORY BLOCK2)에 보관되어 있는 파라미터들을 단말 프로세서 기억장치(MEMORY BLOCK1)로 옮겨오고, 다음에 입력 즉 프랜트에서 측정된 제한량을 읽어들이 제어 연산을 거쳐 출력 즉 프랜트에 공급될 조작량을 발생시킨다. 8개의 제어루우틴을 순차적으로 처리하면 제어동작은 끝난다.

그 결과를 다시 데이터 하이웨이를 통하여 조작자측에 전송하면 한 루우틴이 전부 끝나고 다음 타이밍 신호를 대기하게 된다. 이하 각 부분을 설명하기로 한다.

가. 타이밍 : 속도의 관점에서 보면 타이밍은 분필요하며 제어 루우틴을 연속적으로 회전시키는 것이 유리하지만 일반적으로 제어 대상의 시정수가 긴 것을 고려할 때 고속회전은 무의미하다. 따라서 제어 연산의 편리를 위하여 속도를 약간 희생하고 일정주기로 회전하는 방식을 택하였다. 적분제어와 미

분제어에서 주기를 일정하게 하면 프로그램이 매우 단순해지기 때문이다. 8개 루우틴의 경우면 1회전에 100ms정도의 시간이 필요하며 일반 용도에는 충분한 정도이나 본연구에서 시험 제작한 장치에서는 시중의 저급한 A/D변환기를 사용하였으므로 한주기를 2초로 하였다. 특별한 용도로 주기를 100ms 이하로 줄이려던 고속 A/D변환기를 사용함은 물론이고 프로세서도 16비트 바탕으로 된 것을 이용하는 것이 좋을 것이다.

나. 데이터 교환 : 제어 파라미터를 읽어 들일 때 (그림 3A)와 입출력 데이터를 조작자측으로 전송할 때(그림 3B)의 두 경우에는 프로세서와 프로세서간의 데이터 교환이 필요하게 된다. 두 경우 모두 단말 프로세서에 의한 DMA수법으로 일을 단순화 하였다. 그림 4에 보인 바와 같이 조작자 프로세서는 DMA승인 신호만을 보내고 단절되도록 하였으며 DMA요청이 2회 거듭되어도 승인이 되지 않으면 이를 포기하고 다음 단계로 진행하게 하였는데 이것은 데이터 하이웨이나 조작자 프로세서에 이상이 있을 때에도 당분간은 제어 동작을 계속유지하기 위함이다.

단말 프로세서 단독으로 동작시키고자 할 때 세트 포인드 방식을 사용하면 포텐쇼메타 카드를 데이터 하이웨이에 접속하고 승인 신호를 항상 발생하도록 해주면 좋을 것이다. 그렇게 하면 단말 프로세서의 소프트웨어나 하드웨어를 전혀 변경시키지 않고 사용할 수 있다.

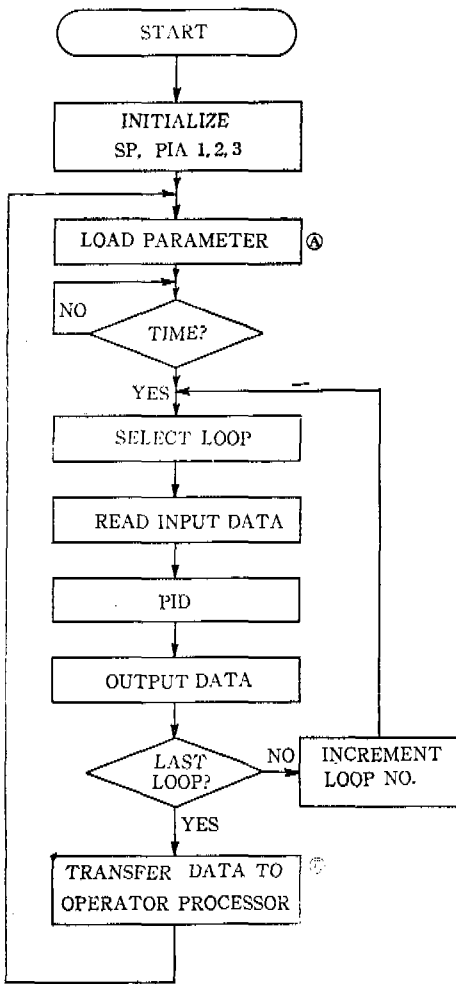


그림 3. 단일 프로세서의 동작 프로우차트

다. 입력 : 입력은 단순히 아나로그 멀티프렉서를 조작하여 해당부우프의 변환량을 A/D변환기 및 PIA를 통하여 읽어 들이던 되는데 시작품에서는 회로 계용변환기를 사용하였으므로 다소 이 과정이 복잡하여 졌다. 속도가 문제되지 않는 경우에는 가격면에서 이 방법도 이용 가능하리라고 생각된다.

다. 제어 연산 : 제어 방식은 비례(P), 적분(I) 및 미분(D) 연산만을 고려하였으므로 이 과정은 직선적으로 처리된다. P, I 및 D의 어떤 조합을 이용할 것인가 하는 것은 단순히 파라미터값에 의하여 판정하도록하였다. 연산식은 식(1)과 (2)를 병렬하여 다음의 형태로 사용하였다(3).

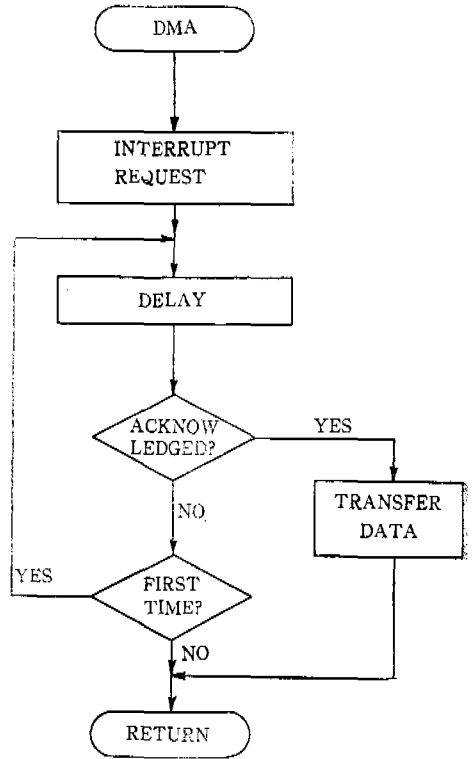


그림 4. DMA 루우틴

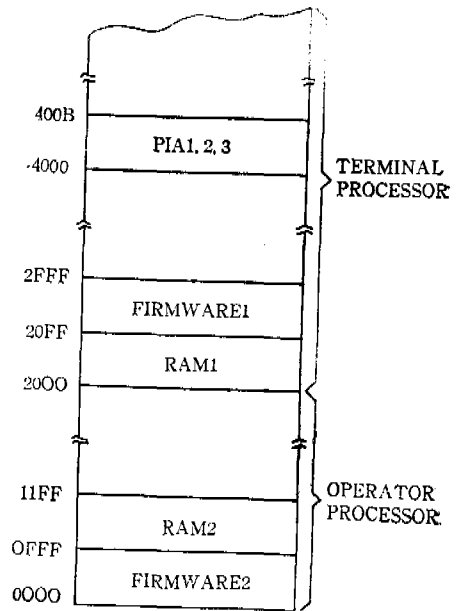


그림 5. 번지할당

$$M(m) = \frac{K_p \cdot E_n + K_i \sum_{i=0}^n E_i D_T + K_D (E_{n-1} - E_n)}{D_T} \quad (3)$$

$$E(n) = R_n - B_n \quad (4)$$

여기서 M, E_n, R_n 및 B_n 은 각각 $m(t), e(t), r(t)$ 및 $b(t)$ 를 주기적으로 샘플링한 값이며 D_T 는 샘플링 주기를 나타낸 것이다. 일정 주기에 의한 운전이 편리하다는 것은 식(3)에서 곧 알 수 있다.

다. 번지 할당 : 번지의 할당은 그림 5와 같다. MPU에서는 PIA들도 기억 소자로 간주되며 2개의 어드레스 선으로 호출된다.

III. 조작자 프로세서

다수의 단말 프로세서들은 데이터 하이웨이를 통하여 조작자 프로세서와 연결된다. 조작자 프로세서가 가져야 할 기능을 열거해 보면 단말측에 대해서는 ①파라메타의 제공, ②입출력 상황의 기록 및 보존, ③운전 상태의 감시, ④이상 진단 및 처리능이 있고, 주변장치와 관련해서 ⑤지시기의 운전, ⑥정시 및 수시 보고 작성, ⑦조작자 키보드와의 교신, ⑧경보의 발생능이 있다. 주 전산기가 있을 때에는 업무의 상당 부분이 이관 되겠으나 ①, ②, ③항의 일부 및 ⑤, ⑥, ⑦, ⑧항은 반드시 담당하여야 할 것이다.

1개의 마이크로 프로세서로써 이 모든 업무를 처리하기에는 다소 무리라고 생각되며 또 이 중에는 구체적 제어 대상과 관련해서만이 파악되는 것이 있으므로 본 연구에서는 단말 프로세서를 유지하기 위한 최소한의 기능인 ①, ⑥ 및 ⑦항만을 구현하기로 하였다. 그림 6에 동작 프로우 차트를 보았으며 기능을 요약하면 다음과 같다.

가. 정시 보고 : 자동적으로 매 시간마다 미리 정해진 형식의 입출력 및 파라메터 상황을 기록해 낸다. 수시 보고 : 그 한 예이다.

나. 수시 보고 : 조작자 키보드의 지령 'R'에 의하여 보고서를 작성하러 할 때는 정시 보고와 같다.

다. 파라메터 변환 : 지령 'P'에 의하여 기억 장치의 파라메터 부분을 열며 그후에는 조작자가 기입하는 데로 파라메터가 기억 장치에 기록된다.

라. 지령 'O' : 작업 중지

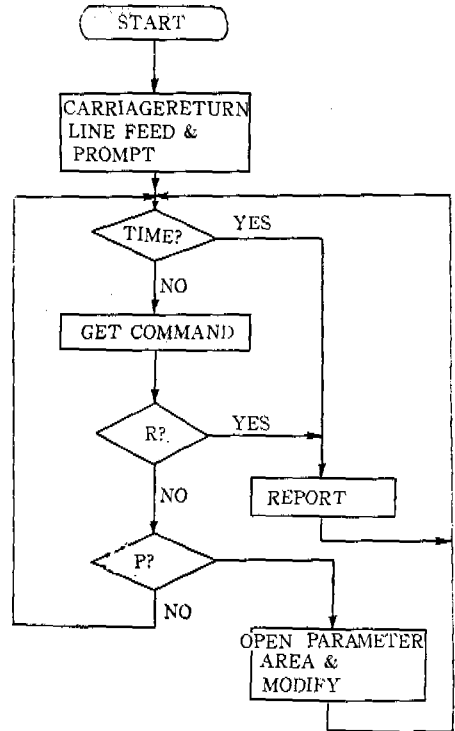


그림 6. 조작자 프로세서의 동작 프로우 차트

TIME 12:00:00

LOOP	REF	KP	TI	TD	INPUT	OUTPUT
00	.324	0.500	10.00	1.00	.331	.455
01	.222	0.800	50.00	1.00	.155	.999
02	.000	0.000	10000	0.00	.000	.000
03	.100	0.235	100.0	0.50	.101	.321
04	.500	2.000	10000	0.00	.510	.100
05	.750	2.500	10000	1.50	.750	.000
06	.450	1.000	60.00	1.80	.450	.055
07	.000	0.000	65535	0.00	.000	.000

그림 7. 정시보고의 예

IV. 하드웨어 구성

기본적 구성은 MPU(Microprocessor Unit) 모듈과 기억장치, 그리고 인터페이스 모듈의 세 부분이며 그림 8에 블록 다이어그램을 보였다. 각 부를 간단히 설명한다.

1. MPU 모듈

MPU 모듈은 프로세서 칩과 클럭 발생기, 버스

라인을 위한 바퍼와 트랜시버, 그리고 어드레스 데코더를 포함한다.

본 연구에서는 MPU1(단말용)에는 MEK6800 D2 KIT를, MPU2(조작자용)에는 SC/MP KIT를 사

용하였으므로 위의 소자들 외에 모니터 ROM, 카세트 테이프 및 키보드용 인터페이스가 추가되어 있어 소프트웨어 개발에 많은 도움을 받았으나 완제품에는 부가된 부분이 불필요하다.

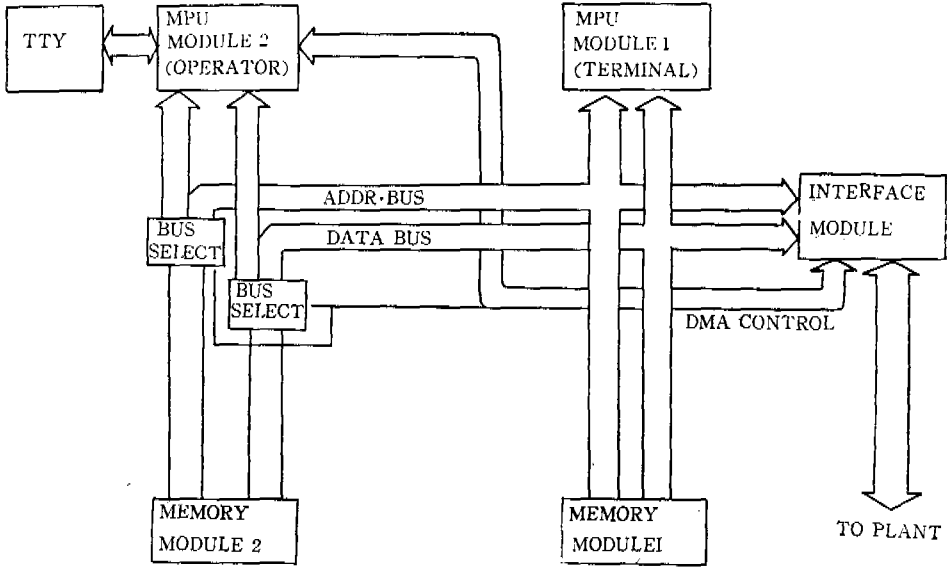
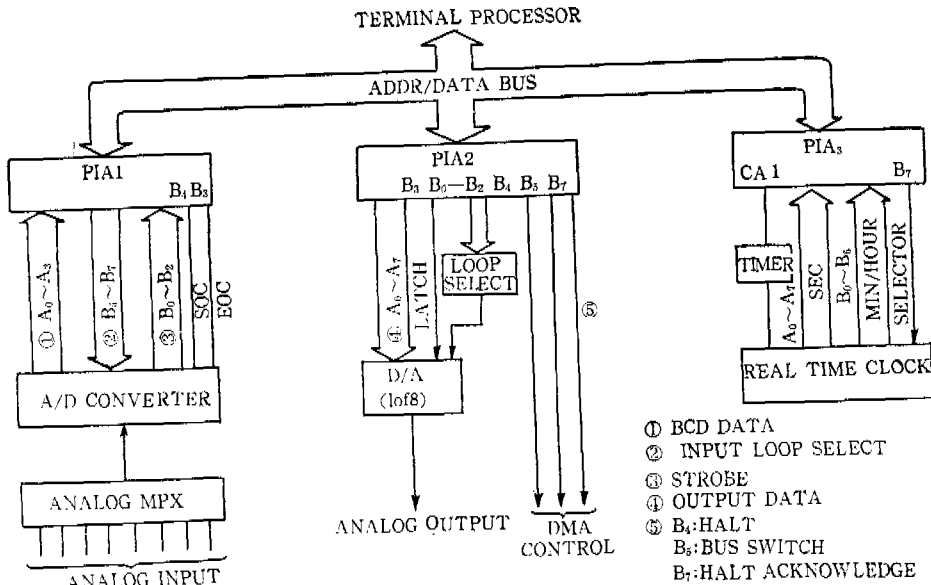


그림 8. 하드웨어 구성도

2. 인터페이스 모듈

인터페이스의 기능은 프래프트와 아나로그 신호의

교환, 각종 콘트롤 신호의 발생 및 접수, 시각 점검의 세 가지인데 그림 9에 보인 바와 같이 세 개의 PIA와 한개의 A/D변환기, 8개의 D/A변환기로



- ① BCD DATA
- ② INPUT LOOP SELECT
- ③ STROBE
- ④ OUTPUT DATA
- ⑤ B₄:HALT
- B₅:BUS SWITCH
- B₇:HALT ACKNOWLEDGE

그림 9. 인터페이스 모듈

구성된다. 여기서 A/D변환기의 변환 속도가 전체의 동작에 치명적 영향을 주므로 선정에 유의하여야 하는데 최근 하이브리드 IC에 의한 여러가지 제품이 소개되고 있으므로 대부분의 목적에 맞는 것을 구할 수 있을 것이다.

본 연구에서는 LD110과 111을 한 조로 하여 사용하였는데 속도가 느리고 데이터들 4회에 나누어 내놓도록 되어 있어 그림 9에서와 같이 스트로브 신호를 이용해야 하는 불편이 있으므로 시정수가 긴 제어대상 외에는 이용하지 않는 것이 좋겠다.

타이밍 인터럽트는 프로세서에 직접 가하지 않고 PIA인터럽트 단자를 이용하였다. 인터럽트가 들어오면 PIA내부 레지스터의 FLAG가 변화를 받으며 이를 데이터 버스를 통하여 점검한다.

콘트롤 신호는 DMA요청, DMA승인, 데이터 및 어드레스 버스 스윙칭의 3개이다.

3. 기억 장치

단말 측과 조작자 측에 각각 2KB의 RAM을 사용하였다. 단말측 기억소자의 대부분은 프로그램이 차지하며 완제품에서 이 부분은 ROM으로 하여야 할 것이다. RAM은 128바이트면 되는데 순간적 전력사고 등을 대비하여 비휘발성의 것을 사용하는 것이 적당하나 시제품에서는 모두 MOS RAM으로 하였다. 조작자 측의 동작 기능을 강화하려면 용량을 8~16KB정도로 증가시켜야 할 것이다.



그림 10. 프로세서와 주변장치

V. 결 론

분산-종합제어 방식에 일맞는 단말용 디지털 프로세서 계통을 구성하고 프로그램을 개발하였다. 몇가지 확인된 사항은 ①아날로그 제어의 개념에서

크게 벗어나지 않으며 동시에 종합제어를 용이하게 할 수 있다. ②소규모 제어 대상은 단독으로 운전할 수 있다. ③범용 마이크로 프로세서로 충분히 제어 계통을 구성할 수 있다. ④프로그램의 크기는 2KB 내외이면 충분하다. ⑤A/D변환기의 성능이 동작에 중요한 영향을 미친다.

디지털 방식에 의한 분산-종합제어는 현재 개발 단계에 불과하나 앞으로 프랜트 제어뿐만 아니라 광범위한 이용 분야가 개척되리라 예상된다. 황무지나 다름없는 국내 제어장치 산업계로서는 디지털 제어는 요원한 느낌이 있으나 장치 구성의 개념이 아날로그 방식과는 다르므로 공기식 및 전자식 장치의 과정을 거치지 않고 직접 개발에 착수할 수 있다고 생각된다.

참 고 문 헌

1. H. Amrehn, "Computer Control in the Polymerization Industry, Automatica", vol.13, pp. 533~545, 1977.
2. M. A. Keyes, "Pulp, Paper and Allied Industry Digital Process Control Systems Status and Trends", Automatica, vol. 13, pp.547~551, 1977.
3. E. H. Bristol, "Designing and Programming Control Algorithms for DDC Systems", Control Engineering, Jan. 1977, pp.24~26.
4. S. J. Bailey, "Process Control '77 : Profile of Profusion", Control Engineering, Aug., 1977, pp.33~36.
5. S. J. Bailey, Direct Digital Control: The On-Line Scene Today, Control Engineering, Jan., 1977, pp.20~23.
6. 渡邊成一. "マイクロコンピュータ應用計装システムオートメーション", vol.22, No.8, pp. 97~101, Aug., 1977.
同, No.9, pp.101~105, Sept., 1977.
同, No.10, pp.101~105, Oct., 1977.
7. M. Reed and H. W. Mergler, "A Microprocessor-Based Control System", IEEE Trans. vol. IECI-24, No.3, pp.253~257, Aug., 1977.
8. Katsuhiko Ogata, "Modern Control Engineering", Prentice-Hall, Inc., 1970.
9. MEK6800 D2 Evaluation Kit II Manual, Motorola Inc., 1977.
10. SC/MP Technical Description, National Semiconductor Co., 1976.
11. M6800 Microprocessor Application Manual, Motorola Inc., 1975.

부록. 단말 프로세서 프로그램

ULSAN INSTITUTE OF TECHNOLOGY MOTOROLA 6800 CROSS ASSEMBLER
 SUPPLIED BY DEPARTMENT OF ELECTRONIC AND ELECTRICAL ENGINEERING
 Loughborough University of Technology England

OUTPUT IN MOTOROLA HEXBUG LOADER FORMAT

NO ERRORS FOUND IN PASS ONE
 NO WARNINGS IN PASS ONE
 45 PERCENT OF SYMBOL TABLE USED

33	20AF							U3	RMB	i
34	20B0							LPNO	RMB	1
35	20B2								ORG	#20B2
36	20B2							TMFX0	RMB	1
37	20B3							TMFX01	RMB	1
38	20B4							TMFX1	RMB	2
39	20B6							TMFX2	RMB	2
40	20B8							TMFX3	RMB	1
41	20B9							TMFX31	RMB	1
42	20BA							TMFX4	RMB	1
43	20BB							TMFX41	RMB	1
44	20BC							TMFX5	RMB	2
45	2100								ORG	#2100
46	2100	CE	4001						LDX	#PIA1AS
47	2103	C6	06						LDAB	#6
48	2105	4F	00						CLR	0.X
49	2107	08						CLR1	INX	
50	2108	08							INX	
51	2109	5A							DECB	
52	210A	26	F8						BNE	CLR1
53	210C	86	00						LDAA	#00
54	210E	B7	4000						STAA	PIA1AD
55	2111	86	F0						LDAA	#F0
56	2113	B7	4002						STAA	PIA1BD
57	2116	86	FF						LDAA	#FF
58	2118	B7	4004						STAA	PIA2AD
59	211B	86	3F						LDAA	#3F
60	211D	B7	4006						STAA	PIA2BD
61	2120	86	00						LDAA	#0
62	2122	B7	4008						STAA	PIA3AD
63	2125	86	80						LDAA	#80
64	2127	B7	400A						STAA	PIA3BD
65	212A	CE	4001						LDX	#PIA1AS
66	212D	86	04						LDAA	#4
67	212F	C6	06						LDAB	#6
68	2131	A7	00					CLR2	STAA	0.X
69	2133	08							INX	
70	2134	08							INX	
71	2135	5A							DECB	
72	2136	26	F9						BNE	CLR2
73	2138	CE	1000					PARALD	LDX	#MEMBL2
74	213B	FF	20B4						STX	TMFX1
75	213E	CE	2000						LDX	#PARAMT
76	2141	FF	20B2						STX	TMFX0
77	2144	FF	20B6						STX	TMFX2
78	2147	C6	40						LDAB	#6A
79	2149	BD	22E3						JBR	TRANS
80	214C	B6	4009					TIME	LDAA	PIA3AS
81	214F	49							ROLA	
82	2150	24	FA						BCC	TIME
83	2152	B6	4008						LDAA	PIA3AD
84	2155	B7	20A0						STAA	SEC
85	2158	86	00						LDAA	#0
86	215A	B7	400A						STAA	PIA3BD
87	215D	B6	400A						LDAA	PIA3B0
88	2160	B7	2061						STAA	MIN
89	2163	86	80						LDAA	#80
90	2165	B7	400A						STAA	PIA3BD
91	2168	B6	400A						LDAA	PIA3BD
92	216B	B7	2062						STAA	HOUR
93	216E	86	00						LDAA	#0
94	2170	B7	20B0						STAA	LPNO
95	2173	CE	20A0						LDX	#IDDATA
96	2176	FF	20B8						STX	TMFX3
97	2179	CE	2000						LDX	#PARAMT
98	217C	FF	20B2						STX	TMFX0
99	217F	CE	2070						LDX	#ERROR
100	2182	FF	20BA						STX	TMFX4
101	2185	CE	20A0						LDX	#BCDIN
102	2188	FF	20BC						STX	TMFX5
103	218B	B6	20B0						LDAA	LPNO
104	218E	0C							CLC	
105	218F	46							RORA	
106	2190	46							RORA	
107	2191	46							RORA	
108	2192	46							RORA	
109	2193	B7	4002						STAA	PIA1BD
110	2196	86	10						LDAA	#10
111	2198	B7	4002						STAA	PIA1BD
112	219B	B6	4002						LDAA	PIA1BD
113	219E	84	08						ANDA	#08
114	21A0	27	F9						BED	EOC
115	21A2	C6	01						LDAB	#1
116	21A4	B6	4002						LDAA	PIA1BD
117	21A7	84	07						ANDA	#7
118	21A9	11							CBA	
119	21AA	26	F8						BNE	STROBE
120	21AC	B6	4002						LDAA	PIA1BD
121	21AF	84	07						ANDA	#0F
122	21B1	A7	00						STAA	0.X
123	21B3	86	04						LDAA	#4
124	21B5	11							CBA	
125	21B6	2F	04						BLE	BINADJ
126	21B8	08							INX	
127	21B9	58							ASLB	
128	21BA	20	E8						BRA	STROBE
129	21BC	FE	20BC						LDX	TMFX5
130	21BF	A6	00						LDAA	0.X
131	21C1	E6	02						LDAB	2.X
132	21C3	58							ASLB	
133	21C4	1B							ABA	
134	21C5	58							ASLB	
135	21C6	58							ASLB	
136	21C7	1B							ABA	
137	21C8	E6	01						LDAB	1.X
138	21CA	58							ASLB	
139	21CB	58							ASLB	
140	21CC	1B							ABA	
141	21CD	1F							TAB	
142	21CE	46							CLRA	
143	21CF	0C							CLC	
144	21D0	58							ASLB	
145	21D1	49							RORA	
146	21D2	58							ASLB	
147	21D3	49							ROLA	
148	21D4	A8	01						ANDA	1.X
149	21D6	58							ASLB	
150	21D7	49							ROLA	
151	21D8	A8	01						ADDA	1.X
152	21DA	46							RORA	
153	21DB	47							RORB	
154	21DC	46							RORA	
155	21DD	47							RORB	
156	21DE	46							RORA	
157	21DF	47							RORB	
158	21E0	FE	20B8						LDX	TMFX3
159	21E3	A7	00						STAA	0.X
160	21E5	E7	01						STAB	1.X
161	21E7	FE	20B2						LDX	TMFX0
162	21EA	A6	00						LDAA	0.X
163	21EC	B7	20A7						STAA	Y0
164	21EF	A6	01						LDAA	1.X
165	21F1	B7	20A8						STAA	Y1
166	21F4	FE	20BA						LDX	TMFX4
167	21F7	A6	00						LDAA	0.X
168	21F9	A7	02						STAA	2.X
169	21FB	A6	01						LDAA	1.X
170	21FD	A7	03						STAA	3.X
171	21FF	FE	20B2						LDX	TMFX0
172	2202	A6	06						LDAA	6.X
173	2204	E6	07						LDAB	7.X
174	2206	FE	20B8						LDX	TMFX3
175	2209	E0	01						SHBB	1.X
176	220B	A2	00						SBCA	0.X
177	220D	FE	20BA						LDX	TMFX4
178	2210	A7	00						STAA	0.X
179	2212	E7	01						STAB	1.X
180	2214	B7	20AA						STAA	XX0
181	2217	F7	20AB						STAB	XX1
182	221A	BD	2327						JSR	MULT16

183	221D	FE	20BC	LDX	TMFX5	233	228F	84	00	ADCB	#0
184	2220	B6	20AC	LDAA	U0	234	2291	EB	03	ADDB	3, X
185	2223	A7	03	STAA	3, X	235	2293	EB	05	ADDB	U0
186	2225	B6	20AD	LDAA	U1	236	2295	FB	20AC	ADDB	U0
187	2228	A7	04	STAA	4, X	237	2298	FE	20B8	LDX	TMFX3
188	222A	FE	20BA	LDX	TMFX4	238	229B	A7	03	STAA	3, X
189	222D	A6	00	LDAA	0, X	239	229D	E7	02	STAB	2, X
190	222F	E6	01	LDAB	1, X	240	229F	B6	20B0	LDAA	LFNO
191	2231	EB	05	ADDB	5, X	241	22A2	80	07	SUBA	#7
192	2233	A9	04	ADCA	4, X	242	22A4	2C	1E	BOE	TRANS2
193	2235	A7	04	STAA	4, X	243	22A6	7C	20B0	INC	LFNO
194	2237	E7	05	STAB	5, X	244	22A9	B6	20B3	LDAA	TMFX01
195	2239	0C		CLC		245	22AC	8B	08	ADDA	#8
196	223A	58		ASLB		246	22AE	B7	20B3	STAA	TMFX01
197	223B	49		ROLA		247	22B1	B6	20B9	LDAA	TMFX31
198	223C	B7	20AA	STAA	XX0	248	22B4	9B	08	ADDA	#4
199	223F	F7	20AE	STAB	XX1	249	22B6	B7	20B9	STAA	TMFX31
200	2242	FE	20B2	LDX	TMFX0	250	22B9	B6	20BB	LDAA	TMFX41
201	2245	A6	02	LDAA	2, X	251	22BC	B6	06	ADDA	#6
202	2247	B7	20A7	STAA	Y0	252	22BE	B7	20BB	STAA	TMFX41
203	224A	A6	03	LDAA	3, X	253	22C1	7E	2185	JMP	SCAN
204	224C	B7	20AB	STAA	Y1	254	22C4	CE	20A0	TRANS2	LDX #10DATA
205	224F	8D	2327	JSR	MULT1A	255	22C7	FF	20B4	STX	TMFX1
206	2252	FE	20BC	LDX	TMFX5	256	22CA	CE	10A0	LDX	#MEM210
207	2255	B6	20AC	LDAA	U0	257	22CD	FF	20B6	STX	TMFX2
208	2258	A7	05	STAA	5, X	258	22D0	C6	23	LDAB	#35
209	225A	B6	20AD	LDAA	U1	259	22D2	8D	2E2	JSR	TRANS
210	225D	A7	06	STAA	6, X	260	22D5	7E	2138	JMP	PARALD
211	225F	FE	20BA	LDX	TMFX4	261	22D8	16		BITIME	TAB
212	2262	A6	00	LDAA	0, X	262	22D9	C4	F0	ANDB	##F0
213	2264	E6	01	LDAB	1, X	263	22DB	84	0F	ANDA	##OF
214	2266	E0	03	SUBB	3, X	264	22DD	54		LSRB	
215	2268	A2	02	SBCA	2, X	265	22DE	1B		ABA	
216	226A	0C		CLC		266	22DF	94		LSRB	
217	226F	47		ASRA		267	22E0	54		LSRB	
218	226C	47		RORE		268	22E1	1B		ABA	
219	226B	B7	20AA	STAA	XX0	269	22E2	39		RTS	
220	2270	F7	20AB	STAB	XX1	270	22E3	37		PSH	
221	2273	FE	20B2	LDX	TMFX0	271	22E4	86	10	TRANS	B
222	2276	A6	04	LDAA	4, X	272	22E6	B7	4006	LDAA	#10
223	2278	B7	20A7	STAA	Y0	273	22E9	86	02	STAA	PIA2BD
224	227R	A6	05	LDAA	5, X	274	22EB	C6	02	LDAA	#2
225	227D	B7	20AB	STAA	Y1	275	22ED	5A		LDAB	#2
226	2280	8D	2327	JSR	MULT16	276	22EE	24	FD	DELAY1	DECB
227	2283	FE	20BC	LDX	TMFX5	277	22F0	4A		AKNW	BNE
228	2286	5F		CLRB		278	22F1	27	33	DECA	DELAY1
229	2287	A6	04	LDAA	4, X	279	22F3	94	80	ANDA	##80
230	2289	AB	06	ADDA	6, X	280	22F5	27	F9	BEQ	AKNW
231	228B	59		ROLB		281	22F7	86	30	LDAA	##30
232	228C	BB	20AD	ADDA	U1	282	22F9	B7	4006	STAA	PIA2BD

283	22FC	C6	60	LDAB	##60	333	236D	7F	20A9	SHIFT	CLR	FF
284	22FE	5A		DECB		334	2370	76	20A7		ROR	Y0
285	22FF	26	FD	BNE	DELAY2	335	2373	76	20A8		ROR	Y1
286	2301	33		PUL	B	336	2376	79	20A9		ROL	FF
287	2302	FE	20B4	LDX	TMFX1	337	2379	77	20AC		ASR	U0
288	2305	A6	00	LDAA	0, X	338	237C	76	20AD		ROR	U1
289	2307	FE	20B6	LDX	TMFX2	339	237F	76	20AE		ROR	U2
290	230A	A7	00	STAA	0, X	340	2382	72	20AF		ROR	U3
291	230C	5A		DECB		341	2385	09			DEX	
292	230D	27	08	BEQ	HESEL1	342	2386	26	B1		BNE	MLP2
293	230F	7C	20B4	INC	TMFX1	343	2388	39			RTS	
294	2312	7C	20B6	INC	TMFX2	344	2389	01		OUT	NOP	
295	2315	20	EB	BRA	TRANS1	345	238A				END	
296	2317	86	10	RESET	LDAA	##10						
297	2319	B7	4006		STAA	PIA2BD						
298	231C	C6	60		LDAB	##60						
299	231E	5A		DELAY2	DECB							
300	231F	26	FD	BNE	DELAY3							
301	2321	86	00	LDAA	#0							
302	2323	B7	4006	STAA	PIA2BD							
303	2326	39		RETURN	RTS							
304	2327	CE	20AC	MULT16	LDX	#U0						
305	232A	4F			CLRA							
306	232B	B7	20A9		STAA	FF						
307	232E	C6	03		LDAB	#3						
308	2330	A7	03	MLP1	STAA	3, X						
309	2332	09			DEX							
310	2333	5A			DECB							
311	2334	2C	FA		BGE	MLP1						
312	2336	CE	0010		LDX	#16						
313	2339	B6	20AB	MLP2	LDAA	Y1						
314	233C	94	01		ANDA	#1						
315	233E	16			TAB							
316	233F	B8	20A9		EORA	FF						
317	2342	27	29		BEQ	SHIFT						
318	2344	50			TSY	B						
319	2345	27	14		BEQ	ADD						
320	2347	B6	20AD		LDAA	U1						
321	234A	F6	20AC		LDAB	U0						
322	234D	8D	20AB		SUBA	XX1						
323	2350	F2	20AA		SCLB	XX0						
324	2353	B7	20AD		STAA	U1						
325	2356	F7	20AC		STAB	U0						
326	2359	20	12		BRA	SHIFT						
327	235B	B6	20AD	ADD	LDAA	U1						
328	235E	F6	20AC		LDAB	U0						
329	2361	BB	20AB		ADDA	XX1						
330	2364	FB	20AA		ADDB	XX0						
331	2367	B7	20AD		STAA	U1						
332	236A	F7	20AC		STAB	U0						

NO ERRORS FOUND IN PASS TWO
NO WARNINGS IN PASS TWO