제 2차년도
최종보고서

사료생산용 인공지능제어 압출성형기의 개발

Development of Animal Feed Extruder with Artificial Intelligence Controller

(단일세부과제로서 중점과제명과 동일함)

동 국 대 학 교

농 림 부

제 출 문

농림부 장관 귀하

본 보고서를 “사료생산용 인공지능제어 압출성형기의 개발” 과제(단일세부과제 “사료생산용 인공지능제어 압출성형기의 개발”)의 최종보고서로 제출합니다.

1997년 12월 29일

주관연구기관명: 동국대학교

총괄연구책임자: 이 승 주

연 구 원: 엄 기 환

요 약 문

사료생산용 압출성형기의 인공지능 제어 시스템을 개발하였다. 제어 하드웨어의 조절부로서 주 제어기(Intel Pentium)와 부 제어기(μPD70208, μPD70325)가 RS-485 통신규약되는 연결되는 시스템을 제작하였다. 쌍축형압출성형기의 공정변수(배럴온도, 구동모터스크류회전속도, 원료공급스크류회전속도, 물공급펌프회전속도, 컷터회전속도)를 계측제어할 수 있고 시스템변수(사출 직전의 다이압력, 사출 직전의 다이온도, SME)를 계측할 수 있도록 온도계, 압력계, 전력계, 유량계, 교류3상모터용 인버터, A/D, D/A 변환기 등을 설치하였다. 모든 센서와 액츄에이터에는 부 제어기가 각각 부착되어 주 제어기와 송수신하면서 주 제어기로부터 명령이 없을 경우에는 기존 상태가 유지되도록 하였다. 또한 제어 패널는 키보드, 주 제어기용 모니터, 비상버튼, 비상용 부저 및 경광등으로 구성하였다.

제어 소프트웨어는 크게 start-up, 정상가동(steady operation), shut-down의 영역으로 구분하여 개발하였다. 제어 알고리즘은 압출성형기로 사료생산을 할 때 측정된 공정변수, 시스템변수, 애완견용 사료압출물의 물성(팽화율, 집단밀도, 수분흡수도)의 상호관계로부터 작성하였다. 이때 시스템변수란 압출물의 물성에 직접적으로 영향을 주는 변수로 알려져 있다. 또한 최종 프로그램은 모니터상에서 window형식으로 보고 입력할 수 있는 graphic interface의 양식으로 개발하였다.

자동 start-up은 크게 2 구간으로 나누어 제어하였다. 먼저 압출물의 온도가 다이쪽의 최종배럴(barrel)의 목표치에 도달할 때까지 히터를 on하고 그 이상이 될 때에는 히터를 off하면서 냉각수 밸브를 on하였다. 이와 같은 과정이 반복이 되면서 압출물의 온도 변화폭이 작아지면 단순히 배럴의 온도를 목표치에 수렴하도록 히터와

냉각수 밸브를 on/off제어하였다. 최종적으로 배럴의 온도 변화폭이 작아지면 정상가동 단계에 도달한 것으로 간주하였다. 이때 구동모터 회전속도, 원료공급속도, 물공급속도(목표치의 3배의 조건에서 시작)는 모두 30초 동안 선형적으로 목표치에 도달하도록 제어하였다. 다이 부근의 압출물 온도가 일정한 값에 도달하기 오래 전에 다른 시스템변수인 압력과 SME는 일정한 값에 수렴한다. 단 구동모터의 power가 극도로 높은 경우가 발생되면 언제라도 비상사태로 간주하여 자동 shut-down의 단계로 넘어가도록 하였다.

자동 정상가동은 압출성형기가 정상가동에 들어간 직후에 측정된 시스템변수를 일정하게 유지하기 위하여 공정변수를 퍼지추론(fuzzy reasoning)방법을 적용하여 제어하였다. 전문가 지식으로서 SME가 증가할 경우에는 물공급속도를 증가시키고, 구동모터회전속도를 감소시키고, 배럴온도를 낮게 제어하였으며, SME가 감소할 경우에는 그 반대 방향으로 제어하였다. 단, 원료공급속도는 모든 경우에 대하여 변화시키지 않았다. 다이부근의 압출물의 온도가 증가할 경우에는 물공급속도를 증가시키고, 구동모터회전속도를 감소시키고, 배럴온도를 낮게 제어하였으며 감소할 경우에는 그 반대 방향으로 제어하였다. 다이부근의 압력이 증가할 경우에는 물공급속도를 증가시키고, 구동모터회전속도를 증가시키고, 배럴온도를 높게 제어하였으며 감소할 경우에는 그 반대 방향으로 제어하였다. 3가지 시스템변수를 동시에 제어하기 위해 상기의 전문가 지식에 관한 퍼지법칙을 작성하였고 각 시스템변수는 퍼지값으로, 공정변수는 확정값(deterministic value)로 설정하였다. 정상가동중 시스템변수가 입력되면 그에 대응하는 공정변수의 제어값들이 퍼지추론을 통하여 출력되어 압출성형기를 제어하는 과정이 반복된다. 이때 퍼지값의 설정은 경험적 지식에 근거하여 임의로 설정하였다.

자동 shut-down에서 배럴의 히터는 shut-down 시작과 함께 off

하고, 물공급속도는 20초내에 초기 목표치의 약 6배까지 선형적으로 증가시켰으며, 구동모터회전속도는 20초후에 3초내에 초기 목표치의 1/6로 감소시킨후 일정하게 유지하였다. 배럴내 충전물의 수분/고형물의 비율을 조절하기 위해 상기와 같이 일단 물공급속도는 단시간내에 최대로 일정하게 해놓고 원료공급속도를 주요 변수로 간주하여 퍼지추론방법을 적용하여 제어하였다. 전문가 지식으로서 막힘현상은 구동모터의 power가 어느정도(fuzzy) 감소할 경우 그 발생 가능성이 매우 높으므로 원료공급속도의 감소를 멈추고, power가 어느정도(fuzzy) 일정할 경우에는 배럴내의 충전물이 밀어내는 힘을 받는데 양호한 상태에 있게 되므로 원료공급속도를 감소하였다. 상기의 전문가 지식에 관한 퍼지법칙을 작성하였고 시간 간격을 달리한 power의 변화량은 퍼지값으로, 원료공급속도는 확정값(deterministic value)로 설정하였다. Shut-down중 power가 입력되면 퍼지추론을 통하여 그에 대응하는 원료공급속도가 출력되어 압출성형기를 제어하는 과정이 반복된다. 이때 퍼지값의 설정은 경험적 지식에 근거하여 임의로 설정하였다.

SUMMARY

An artificial intelligence control system for animal feed extruder was developed. Controller consists of main controller(Intel pentium) with several subcontrollers(μPD70208, μPD70325) communicated in RS-485 protocol. Thermocouple, pressure gauge, power gauge, flow gauge, inverter for 3 phase AC motor, A/D, D/A converters, etc. were used to measure and control the process variables(barrel temperature, main motor screw rpm, feeder rpm, water pump rpm, cutter rpm) and the system variables(pressure and temperature before die, SME) of twin screw extruder. Subcontrollers were individually assigned to sensors and actuators and communicated with main controller. When no signal from main controller is transferred, subcontrollers keep actuators at already existing conditions. Control panel contained keyboard, monitor for main controller, and emergency button, buzzer and lamp.

Control software was developed for three operation stages, i.e., start-up, steady operation and shut-down. Software algorithm was made on the basis of relationships between process variables, system variables and physical properties of pet food(expansion ratio, bulk density and water absorption amount). The system variables were known to have high correlation with extrudate physical properties. Computer program was constructed as an window graphic interface of input and output between operator and control system.

Start-up was controlled by two stages. First, until extrudate

temperature before die reaches at set point of barrel temperature, barrel heater is on. After that, heater is off and barrel cooler is on. As such process is repeated, the change of extrudate temperature gets little. Second, after the extrudate temperature is kept contant, the heat and cooler are simply on/off-controlled to set point. Finally, the temperatures of extrudate and barrel become constant, and start-up is all set. During start-up, main motor rpm, feeder rpm, water pump rpm(initially starts with 3 times higher than set point) are linearly increased to set point in 30 sec. In case that main motor power exceeds over max. load, start-up automatically turns to shut-down stage.

Steady operation after start-up was controlled by fuzzy reasoning to keep system variables constant. Expert knowlege was obtained as follows. As for increase in SME, water pump rpm is controlled to increase, main motor rpm decrease, and barrel temperature decrease. As for decrease in SME, the process varibles above are controlled in opposite direction. However, feeder rpm is fixed as constant in all cases. As for increase in extrudate temperature before die, water pump rpm is controlled to increase, main motor rpm decrease, and barrel temperature decrease. As for decrease in extrudate temperature, the process varibles above are controlled in opposite direction. As for increase in pressure before die, water pump rpm is controlled to increase, main motor rpm increase, and barrel temperature increase. As for decrease in pressure, the process varibles above are controlled in opposite direction. In order to control three system variables above simultaneously, fuzzy reasoning was adopted with fuzzy rules composed of fuzzy values of system varibles as if-part and deterministic values of process variables as then-part. If change in system variables is inputted to fuzzy reasoning, change in process variables are outputted to control. Such process was repeated.

In shut-down, barrel heater is initially off. Water pump rpm is linearly increased to 6 times higher than existing set point in 20 sec, then main motor rpm is linearly decreased to 1/6 lower than existing set point in 3 sec. As expert knowledge for feeder rpm control, a blockage during shut-down occurs when main motor power rapidly decrease. Accordingly, feeder rpm should be decreased only when avoiding rapid decrease in main motor power. Fuzzy reasoning was adopted with fuzzy rules composed of fuzzy value of change in main motor power as if-part and deterministic value of feeder rpm as then-part. If change in main motor power is inputted to fuzzy reasoning, change in feeder rpm is outputted to control. Such process was repeated.

CONTENTS

목 차

제 1 장 서 론

제1절 연구개발의 목적

압출성형기(extruder)가 1940년대에 사료 산업에 소개된 이래, 최근 15년간에 걸친 압출성형 기술의 개발과 응용은 실로 대단하였다. 그 결과로 인하여 압출성형 기술은 동물 사료 제조에 있어서 점차 경제적인 열처리 가공 방법으로 각광 받게 되었다. 오늘날 애완동물 사료 제조에서는 압출성형기가 없으면 안 되는 실정이며 양식어류용 사료 제조에 있어서도 압출성형 가공은 급속하게 신장하고 있을 뿐 아니라 소, 돼지, 닭 등의 가축용 사료 제조, 그리고 사료용 원료 가공에 이르기 까지 압출성형기의 응용 분야는 날로 확대되고 있다.

국내 사료제조업체의 압출성형기 보유 현황을 보면 약 20개 업체가 약 40대의 압출성형기를 사용하고 있는데 압출성형기의 복잡한 부품 구조, 다양한 원료의 구성에 따른 가공, 가열 중의 물성 변화에 따른 종합적인 이해 부족 등으로 인하여 최초 기계 공급업자가 제시하는 경험적 자료에 주로 의존하여, 사료 제조업자 자신이 압출성형기의 성능을 파악, 활용 기술을 개발하는 등에 머물러 있는 실정이다. 한편 최근 기계의 자동화가 급속도로 진행되고 있는데 미래의 첨단 기술인 컴퓨터 인공지능제어방식을 압출성형기에 적용한다면 상기와 같은 생산 현장에서 압출성형기 활용 및 가동 조건에 대한 이해 부족 등으로 파생되는 문제점을 충분히 극복할 수 있을 것으로 생각된다.

일반적으로 사료나 식품생산용 압출성형기를 올바르게 가동하는 데는 상당한 시간과 경험이 요구되는 것으로 알려져 있다. 압출성형

기의 가동은 3단계로 이루어지는데, 먼저 기계가 무리가 가지 않도록 서서히 가동을 시작하여(start-up), 압출성형기의 가동이 정상상태에 이르게 되면 비로서 압출물을 생산하기 시작한다(steady operation). 압출성형기의 가동을 마치기 위해서 서서히 가동을 종료하는데(shut-down), 상기의 3단계 가동은 숙달된 전문가만이 할 수 있다. 따라서 본 연구에서는 각 단계에 대하여 인간 전문가를 대치할 수 있는 컴퓨터 인공지능제어 시스템을 아래와 같이 개발하였다.

Start-up에서 정상가동까지 이르는 시간이 짧지 않기 때문에 그 동안 나오는 압출물 쓰레기가 많게 된다. 또한 시작을 너무 빨리 진행시키면 기계에 무리가 생겨 압출성형기의 구동 모터가 손상되거나 또한 압출물이 나오다 멈추게 될 수 있기 때문에 이를 해결할 수 있는 자동제어 시스템을 개발하고자 하였다.

정상가동 중에는 아무리 시작을 잘하여 기계의 작동조건이 잘 맞춰져 있다고 할지라도 예기치 못한 변화에 의해 나오던 압출물의 물리적 상태가 갑자기 변할 수 있다. 즉, 정상상태에서 압출물을 이상없이 생산을 하고 있는 도중이라도 원료의 투입량, 원료 자체의 수분함량, 원료의 지방함량 등이 불균일할 경우에는 생산 도중 품질이 다른 압출물이 섞여 나오게 된다. 이상과 같은 경우를 방지하기 위해 원료 등의 성질이 불균일하다고 할지라도 항상 같은 품질의 압출물이 생산되도록 할 수 있는 자동제어 시스템을 개발하고자 하였다.

압출성형기의 사용을 중지하기 위해 작동조건을 잘 못 조절하면 압출물의 흐름이 기계 내에서 막히게 되어 결국 압출성형기를 분해해야 하는 등 큰 무리가 생기게 된다. 또한 shut-down시간이 너무 오랫동안 경과하면 역시 그간 버리게 되는 압출물의 쓰레기가 많아지게 된다. 따라서 이를 해결할 수 있는 자동제어 시스템을 개발하고자 하였다.

제2절 연구개발의 범위

사료 생산용 압출성형기의 공통적인 문제인 자동화를 아래와 같은 개별적인 목표를 종합화함으로서 완성하고자 하였다.

- 압출성형기 가동시 데이타 수집 및 제어 하드웨어를 개발한다.
- 압출성형기를 이용한 사료 생산의 공정 조건을 확립한다.
- 제어 하드웨어를 가동시킬 인공지능제어 소프트웨어를 개발한다.
- 제어 내용으로 압출성형기의 자동 start-up, 정상가동, shut-down, 비상 사태 및 jamming시 자동 shut-down 등을 포함한다.

구 분	연 구 개 발 의 범 위
1차 년도 (1995.12.- 1996.12)	- 제어 시스템의 하드웨어 : 계측부, 조절부, 조작부 - 압출성형기에 의한 사료 생산 조건 : 압출성형공정변수가 시스템변수에 미치는 영향; 시스템변수가 압출물의 물성에 미치는 영향
2차 년도 (1996.12.- 1997.12)	- 제어 시스템의 하드웨어 : 보완 완성 - 제어 시스템의 소프트웨어 : 생산 조건으로부터 전문가 지식의 구축; 자동 start-up, 정상가동, shut-down의 인공지능제어 알고리즘의 완성; 최종 컴퓨터 프로그램의 완성; 사료생산 인공지능제어의 case studies

제 2 장 압출성형기 제어 하드웨어의 개발

제1절 서 설

사료 생산용 인공지능 제어 압출성형기의 자동화를 위한 하드웨어의 개발은 디지털 제어방식인 컴퓨터에 의한 제어방식을 이용한다.

컴퓨터에 의한 제어방식으로는 기존의 방식인 센서(sensor)와 조작기(actuator)를 그대로 한 대의 컴퓨터에 연결하여 제어하는 방식과, 최근의 방식인 컴퓨터 통신에 의한 멀티 프로세서를 이용하는 방식이 있다. 본 연구에서는 분산제어 및 컴퓨터 종합생산 시스템을 위하여 후자를 선정하였다.

컴퓨터 통신에 의한 멀티 프로세서를 이용하는 방식에는 각각의 센서와 조작기에 마이크로 프로세서가 각각 있으므로 정밀 제어가 가능하고, 메인(main) 프로세서의 일을 분담하게 되어 전체 제어기의 효율이 증대되어 제어속도가 상승하고 시스템간의 배선의 양이 현저하게 줄어들어 배선간의 간섭으로 인한 오차가 줄어들므로 인공지능 알고리즘을 이용하기에 적합하다. 또한 마이크로 프로세서(micro- processor)가 고장이 날 경우, 한 대의 컴퓨터에 의존하는 것보다 여러대의 마이크로 프로세서에 분산하여 제어를 수행 한다면, 전체 시스템이 오류를 범할 확률이 줄어들게 된다.

본 연구에서는 멀티 프로세서를 이용하는 제어용 하드웨어를 다음과 같은 순서로 개발 하였다.

1. 전체 하드웨어를 블록으로 나누고, 각 블록에 알맞은 마이크로 프로세서의 수준을 결정하여 실제 사용할 마이크로 프로세서를 선정

한다.

2. 통신의 전기적 특성과 프로토콜(protocal) 방식 그리고 데이터 블록을 결정하고, 결정된 통신 방식의 하드웨어를 설계하고 제작 실험한다.

3. 각각의 나누어진 블록을 설계하고 제작 실험하여 각 블록을 완성한다.

4. 나누어진 불록들을 한데 모아 통신과 제어 실험을 하여 각 블록의 잘못된점을 찾아 수정한다.

5. 실제 하드웨어를 구성하여 실험한다.

제2절 블록설정과 마이크로 프로세서 선정

1. 전체 하드웨어의 블록설정

전체적인 하드웨어를 구성하기 위하여 먼저 하드웨어를 각각의 블록으로 나누어 설계하였으며, 나누는 기준은 다음과 같다.

가. 같은 동작을 하고, 한곳으로 모을 수 있는 것은 하나의 블록으로 한다.

나. 같은 동작을 하지만 각각의 기준이 틀리거나 많은 데이터 처리가 요구되고 제어량이 많은 경우에는 각각을 분리한다.

다. 전체 시스템의 가격을 낮출수 있는 방향으로 분리한다.

이와 같은 기준으로 나누어진 전체 블록선도는 Fig.2-1과 같다.

Fig.2-1 Block diagram of complete system

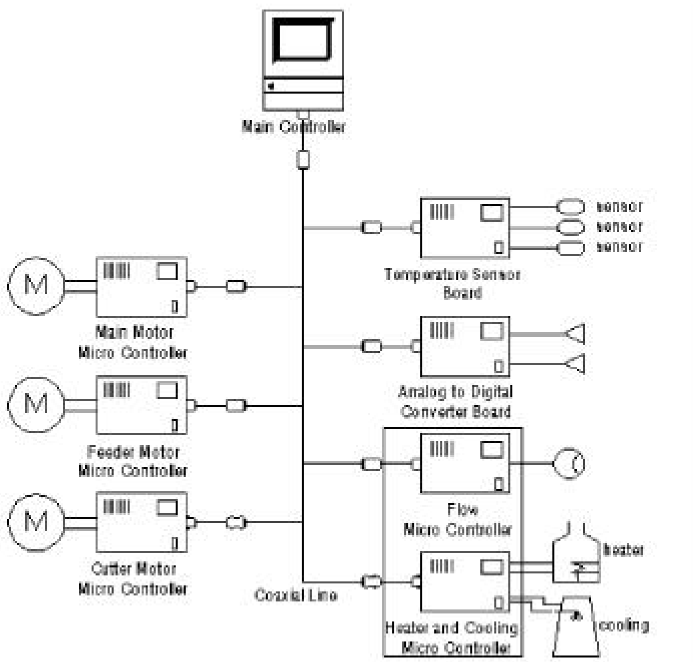
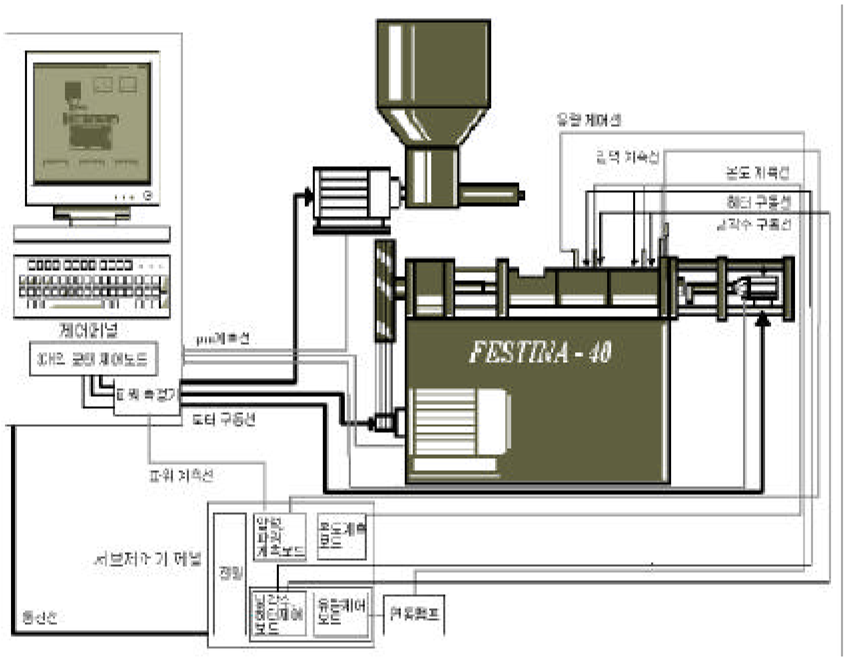


Fig.2-1의 전체적인 시스템에서 주 제어기(Main Controller)는 조절부와 표시부의 역할을 하며, 모터 제어기(Main Motor Controller)는 조절부와 조작부의 역할을 동시에 한다. 온도측정보드와 아날로그 신호를 측정하는 보드는 계측부의 역할을 하고, 히터, 냉각수, 유량 제어기는 조작부로서의 역할을 하게 된다. 이와 같이 구성한 전체 시스템은 Fig.2-2와 같다.

Fig.2-2 Block diagram of complete control system

2. 마이크로 프로세서의 선정



가. 주제어기의 선정

주제어기의 선정은 다음의 기준으로 선정하였다.

(1) 우선 유지 보수가 쉬워야 한다.

(2) 인공지능 알고리즘을 수행할 수 있을 정도의 성능이 요구된다.

(3) 인공지능 알고리즘의 이식이 쉬워야 한다.

(4) 사용자가 쉽게 조작할 수 있어야 한다.

(5) 충분한 통신성능이 요구된다.

(6) 많은 양의 데이터를 임시 저장할 수 있어야 한다.

이와 같은 기준에 의해 주 제어기는 Intel Pentium Processor를 사용한 PC급의 제어기를 선정하였다. 초기에는 DSP Processor를 사용한 소형 마이크로 프로세서와 LCD 장치를 사용하려 했으나 Low level의 언어를 사용해야 하므로 인공지능 알고리즘의 이식과 유지 보수가 매우 어렵고, 소형 마이크로 프로세서는 많은 양의 데이터를

임시 저장할 경우 가격이 Intel Pentium Processor를 사용한 PC급의 제어기 정도로 올라가며, LCD 장치와 키 패드로는 PC급의 제어기보다 사용자가 쉽게 조작할 수 없기 때문에 이를 선정하게 되었다.

나. 모터 제어기의 선정

모터 제어기의 프로세서 선정은 다음의 기준으로 선정하였다.

(1) 제어프로그램의 개발과정에서 간편한 수정과 간단한 프로그램 이식성의 특성이 있어야 한다.

(2) 프로세서내에 정확한 통신특성과 여러 잠재기능을 갖을 수 있도록 다량의 외부 포트를 갖고 있어야 한다.

(3) 통신과 함께 외부에서 rpm값이 계수되고, 제어알고리즘이 동시에 처리될수 있을 정도의 속도를 갖고 있어야 한다.

(4) 개발환경이 우수한 특성이 있어야한다.

이와 같은 기준으로 모터 제어기에 사용되는 프로세서로 μPD70325을 선정하였다.

μPD70325마이크로프로세서는 NEC사의 16bits 프로세서로 원보드로 설계하여 가격이 저렴하고, 24bits의 외부 port와 1byte의 comperator, 2 채널의 직력통신 포트, 그리고 3개의 외부 인터럽트 등, 다양한 기능을 가지고 있어 PC를 통한 개발환경에서 쉽게 프로그램과 통신을 통한 실행에의해 알고리즘을 구현할 수 있다.

다. 온도 센서 보드의 프로세서 선정

온도 센서 보드의 프로세서 선정은 다음의 기준으로 선정하였다.

(1) 자체 오차 보정 알고리즘을 갖는 정확한 통신이 가능해야 한

다.

(2) 온도 센서의 특성에 따라 비선형인 온도 측정 곡선을 선형화할 수 있는 알고리즘을 수행할 수 있는 성능이 있어야 한다.

(3) 미세한 온도 측정 값에 여러 곳에서 노이즈(noise)가 들어오므로 이를 제거 할 수 있는 필터 알고리즘을 수행할 수 있는 성능을 가져야 한다.

(4) 전체 회로의 크기가 작고 가격이 저렴해야 한다.

이와 같은 기준에 의해 온도 센서 보드의 프로세서는 μPD70208으로 선정하였다.

이 프로세서는 현재 페이저 중계기나 팩스를 제어하는데에 많이 사용되는 프로세서로 가격이 저렴하고 성능이 뛰어나며 제작할 경우 부피가 매우 작은 특징을 가지고 있다.

라. 아날로그 신호(압력, 파워, 유량) 측정 프로세서 선정

아날로그 신호를 측정하는 보드의 프로세서 선정은 다음의 기준으로 선정하였다.

(1) 자체 오차 보정 알고리즘을 갖는 정확한 통신이 가능해야 한다.

(2) 미세한 측정 값에 여러 곳에서 노이즈가 들어오므로 이를 없앨 수 있는 필터 알고리즘을 수행할 수 있는 성능을 가져야 한다.

(3) 전체 회로의 크기가 작고 가격이 저렴해야 한다.

(4) 여러 가지의 아날로그 신호를 동시에 처리할 수 있는 성능이 요구된다.

이와 같은 기준에 의해 아날로그 신호를 측정하는 보드의 프로세서는 온도 센서 보드의 프로세서와 같은 μPD70208으로 선정하였다.

마. 유량, 히터, 냉각수 제어 프로세서 선정

유량, 히터, 냉각수 제어 프로세서 선정은 다음의 기준으로 선정하였다.

(1) 보정 알고리즘을 갖는 정확한 통신이 가능해야 한다.

(2) 아날로그값을 측정할 수 있어야 한다.

(3) 전체 회로의 크기가 작고 가격이 저렴해야 한다.

이와 같은 기준에 의해 유량제어 보드의 프로세서는 μPD70325으로 선정하였다.

제3절 통신방식의 결정

여러대의 컴퓨터 시스템이나 제어기들을 통신 회선으로 접속하고, 각종 정보를 서로 교환 할 수 있도록 한 것을 네트워크(network)라고 한다. 그 이용 형태는 일반적으로 정해진 규약을 가지고 있는 것도 있고, 또 그렇지 않은 것도 있어 일괄적으로 종합할 수 없다. 또한 그 결합 형태도 여러 가지 이다.

이들 네트워크의 주된 목적은

1. 자원의 공용(주로 프로그램이나 데이터)
2. 부하의 분담
3. 신뢰성 향상 등 이다.

통신에는 병렬 통신과 직렬 통신이 있는데 병렬 통신은 속도가 빠르다는 장점이 있으나 케이블의 수가 매우 증가하며 1m이상의 통신은 거의 불가능하다는 단점이 있어 산업 현장이나 일반적으로는 많이 사용하지 않는다. 직렬통신은 병렬통신보다 느리다는 단점이 있

으나 최소 2개의 선으로 통신이 가능하며 원거리 통신이 가능하여 실제 산업현장이나 일반인들의 통신에 많이 사용된다. 그러므로 본 연구에서는 직렬 통신의 방법을 사용한다.

직렬통신은 여러 가지로 분류를 할 수 있으나 대표적인 것을 보면 동기통신 방법과 비동기 통신 방법이 있다.

동기통신 방법은 데이터 블록을 한번에 동기를 잡기 때문에 비동기 통신보다 속도가 매우 빠르나 동기신호를 놓쳤을 경우에는 많은 양의 데이터를 잃어버리는 문제가 있어 거리가 매우 짧은 직렬메모리나 하나의 하드웨어에서 멀티 프로세서를 사용할 경우 고속의 데이터 공유를 위해 사용한다. 비동기 통신은 동기통신에 비해 속도는 느리나 데이터로 동기를 잡기 때문에 하나의 데이터가 전송의 기본 단위가 된다. 또한 클럭(clock)도 내부에서 설정할 수 있으므로 가장 적당한 통신방법으로 볼 수 있다. 그러므로 본 연구에서는 비동기 통신 방법을 사용한다.

비동기적인 직렬 통신을 위해서는 실제 데이터를 주고받는 전기적 규약이 필요하다. 전기적 규약은 여러 가지가 있으나 일반적으로 제일 많이 사용하는 RS-232C와 산업현장에서 많이 쓰이는 RS-485의 전기적 규약은 Table 2-1과 2-2이다.

Table 2-1. A electorical rule for RS-232C

송신측	최대 출력전압	±25V
	최소 출력전압	±5V
	최소 출력저항	300Ω
	최대 출력전류	500㎃
	최대 스위칭 속도	30V/μs
수신측	입력저항	3 7Ω
	입력 드레시홀드	±3V
	입력전압	±25V
*실제 접속에서는 ±5V ±15V이어서 이 규격에 맞추어야 한다.		

Table 2-2. A electrical rule for RS-485

송신측	최대 차동 출력전압	5V
	최소 차동 출력전압	1.5V
수신측	입력저항	12kΩ
	입력 차동 드레시홀드	±0.2V
	입력 hysteresis	70㎷

압출성형기에 사용할 수 있는 전기적 규약은 다음의 기준에 의해 선정하였다.

1. 대형 모터를 구동시키므로 많은 노이즈가 발생하므로 노이즈에 강해야 한다.
2. 프로그램이 간단히 구현될 수 있어야 한다.
3. 하드웨어가 간단해야 한다.
4. 배선의 양이 적어야 한다.
5. 충분한 속도가 보장되어야 한다.

이와 같은 기준에 의해 변형된 RS-485의 전기적 규약을 선택 하였다.

RS-232C의 방식은 하드웨어가 간단하고 많은 곳에 사용되어 프로그램이 매우 쉬우나 다른 방식에 비하여 노이즈에 약하고 압출성형기 제어에 사용할 경우 그 배선의 양이 서브 프로세서와 메인 프로세서간에 각각 최소 3개의 배선이 필요하므로 제외 하였다. 또한

RS-485방식은 배선의 양이 현저히 줄어들고 하드웨어가 간단하나 여러 가지 부가적인 동작과 타이밍(timing) 문제로 인하여 일부 제어기에서 프로그램이 복잡하고 어려운 문제점이 발생하고 또한 선로상의 임피던스를 맞춰주어야 하므로 변형된 RS-485의 전기적 규약을 선택 하였다. 변형된 RS-485의 전기적 규약은 RS-485와 RS-232C의 중간적인 특징을 가지고 있어 RS-485와 같은 버스형태의 연결이 가능하고 하드웨어로의 구현이 간단하며 RS-232C의 프로그램을 조금만 수정하여 사용할 수 있는 장점이 있다. Fig.2-2 는 변형된 RS-485의 인터페이스 회로도 이다.

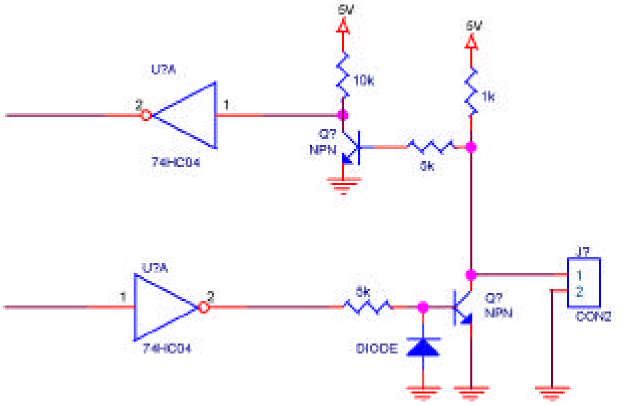


Fig.2-2 Schematic diagram of modified RS-485 interface circuit

RS-485와 변형된 RS-485의 실험결과 RS-485방식이 조금더 노이즈에 강한결과를 얻었으나 압출성형기에서 사용할때에는 변형된 RS-485방식과 비슷하게 노이즈가 많이 생성되어 동축선로를 사용해야 하는 결과를 얻을 수 있었다.

통신 프로토콜은 여러 가지가 있으나 CSMA/CD 방식과 컨트롤러

에 의한 관리형의 방식이 산업현장에서 주로 많이 사용되고 있다. CSMA/CD 방식은 디바이스(device)들이 항상 선로를 확인하여 복수의 디바이스들이 송신했을 경우의 충돌을 감지하고 수정하여 다시보내는 형식으로 한 바이트 단위의 전송시에는 매우 높은 효율을 가진다. 그러나 프로그램이 복잡하고, 압출성형기와 같이 블록단위로 송수신을 할 때에는 오히려 통신 라인의 효율이 크게 떨어지게 된다. 이는 한 블록을 모두 전송받을 때 까지는 충돌을 예측할 수 없기 때문이다.

컨트롤러에 의한 관리형의 방식은 하나의 컨트롤러가 송신을 허가하는 방식으로 충돌의 발생확률은 극히 미미하고, 통신효율은 떨어지나 블록을 단위로 송 수신을 하였을 경우 충돌 확인과정이 없으므로 통신 효율이 증가될 수 있다. 그러므로 본 연구에서는 컨트롤러에 의한 관리형의 방식을 사용하여 주 제어기가 송신을 허가하게 한다.

통신 속도는 빠를수록 좋으나 일부 사용한 서브 제어기의 성능에 의해 4800bps의 속도로 제한 되었다. 실험을 통하여 이정도의 속도로도 충분함을 확인 하였다.

송 수신의 데이터는 한 바이트를 사용하지 않고 여러 바이트를 묶어 블록으로 사용한다. 이는 RS-485의 전기적 규약에 의해 여러개의 서브 제어기와 주제어기가 버스형태로 연결되어 있기 때문에 각각의 프로세서를 구분해 줄 주소들이 데이터와 함께 전송되기 때문이다.

데이터 블록의 한 바이트는 1개의 시작 비트, 8개의 데이터 비트 1개의 스톱비트(stop bit)로 구성된다. 데이터 블록은 8비트(1 바이트)의 단위로 나누어져 있으므로 다루기가 쉽다. 데이터 블록은 Table 2-3과 같다.

Table 2-3. Data block type

start byte
transmit address byte
receive address byte
command byte
high data byte
low data byte
check sum
stop byte

각 바이트의 포맷은 Table 2-4.**와 같다**.

Table 2-4. Format of data block

바이트 명	할당 값(2진 bit)	비고
start byte	01010101	
address byte	00010000	main PC or main Controller
	00000001	motor 1 controller
	00000010	motor 2 controller
	00000011	motor 3 controller
	00000100	온도 센서 보드
	00000101	압력등의 아날로그 센서 보드
	00000110	히터, 냉각수 온-오프 제어 보드
	00000111	유량 제어 보드
command byte		표 3 참조
data high byte		데이터를 16비트로 주고 받기위한 영역
data low byte		
check sum	address 'XOR' command 'XOR' data high 'XOR' data low	통신 오류의 보완을 하기위한 것임
stop byte	10101010 (bit)	

Table 2-5는 Table 2-4의 포맷중 명령어 대한 표 이다.

Table 2-5. Command format of data block

서브 제어기	할당 값(2진 bit)	비고
motor 1 controller	00000010	rpm 설정
	00000011	rpm 계측
motor 2 controller	00000010	rpm 설정
	00000011	rpm 계측
	00000100	인버터 구동
	00000101	인버터 정지
motor 3 controller	00000010	rpm 설정
	00000011	rpm 계측
	00000100	인버터 구동
	00000101	인버터 정지
	00000110	긴급상황 표시
온도 센서 보드	00000001	초기화
	10000000	1번 온도 계측
	10000001	2번 온도 계측
	10000010	3번 온도 계측
압력등의 아날로그 센서 보드	00000001	초기화
	10010000	1번 전류 계측
	10010001	2번 전류 계측
	10010010	3번 전압 계측
	10010011	4번 접압 계측
히터, 냉각수 온-오프 제어 보드	00010000	on-off 명령
	00010001	on-off 상태 계측
유량 제어 보드	00010000	유량 설정
	00010001	유량 계측

RS-485의 전기적 특성에 의한 메인 프로세서에서 어떤 명령어들을 주게되면, 주었던 명령이 자기에게 돌아온다. 그러므로 돌아온 명령을 비교하여 통신 오류가 일어났는가를 확인할 수 있다. 그리고 각 제어보드나 센서보드에서 데이터를 얻을때는 먼저 데이터를 달라는 명령을 먼저 보내야 한다.

제 4 절 서브(sub) 블록의 설계, 제작 및 실험

1. 주 제어기의 설계, 제작 및 실험

주 제어기는 각 서브 제어기에서 데이터를 받아 인공지능 알고리즘이나 사용자가 설정하는 임의의 제어명령에 의해 전체 제어 데이터를 만들어 각 서브 제어기에 보낸다. 서브제어기는 주 제어기가 데이터를 요청했을 경우 가장 최근의 데이터를 전송하여 주고 주 제어기가 보낸 제어 데이터에 따라 정확히 동작을 한다. 그러므로 중간에 정상적인 동작을 하고있을 경우 주 제어기가 꺼져 있어도 각 서브 제어기들은 이전의 제어 데이터를 가지고 계속적으로 동작을 하게 된다.

주 제어기는 주로 인공지능 알고리즘을 수행, 외부와의 통신, 그리고 유저(user) 인터페이스의 제공을 전담하게 하였다. 주 제어기는 그 자체로 전체적인 제어가 가능하게 해야 하므로 제어 패널을 필요로 한다.

인공지능 알고리즘을 수행하기위해 다른 보조제어기보다 성능이 좋은 프로세서를 사용하였고, 한번 세팅된 값을 계속하여 사용할 수 있도록 내부에 데이터를 저장할 수 있게 하였다. 주 제어기 패널에는 유저 인터페이스를 제공하기위해 키보드와 모니터가 있고, 자동제어에 맞게 외부로 보이는 버튼을 최소한의 갯수를 사용하였다. 버튼은 전체 시스템을 끌 때(off)에만 사용하게 하였고, 그외에는 모두 키보드를 통해 입력을 받게 하였다. 모든 제어 변수는 수동으로 키보드에 의해 입력될 수 있게 하였다. 사람이 멀리 떨어져 있을 경우에도 사람에게 긴급사항을 알릴 수 있도록 부저(Buzzer)와 경광등을 설치하였다. 외부와의 통신은 RS-485통신규약을 가지게 설계 하였다. 외부와의 통신은 멀리 떨어져 있는 곳에서 주제어기를 제어해 압출성형기를 동작 시킬 수 있게 하여준다. 그러므로 CIM(Computer Integrated Manufacturing)을 가능하게 된다. Fig.2-

3은 제작한 주 제어기의 사진이다.

Fig.2-3 Photograph of main controller

2. 모터 제어기의 설계, 제작 및 실험

압출 성형기에서 사용되는 모터는 3상 유도 전동기로 15마력의 main 모터, 1마력의 feeder 모터, 1마력의 cutter 모터가 사용되었다. 그리고 각각의 모터를 제어하기위해서 PWM인버터가 사용이 되었는데 main 모터에는 11[Kw]의 PWM인버터가 사용이 되었고, feeder 모터와 cutter 모터에는 각각 1[Kw]의 인버터를 사용하였다.

Fig.2-4는 인버터와 제어알고리즘을 연결해주는 interface card로 AC모터의 rpm값과 설정값을 전송하는 기능을 한다. 실험에서 사용된 rpm값은 16bit로하여 계측값의 정밀성을 확보하였다.

Fig.2-4 Interface Card

Fig.2-5는 실험제작한 인버터의 controller로 μPD70325를 사용하여 구성하였다.

압출성형기를 동작 시킬 수 있게 하여준다. 그러므로 CIM(Computer Integrated Manufacturing)을 가능하게 된다. Fig.2-3은 제작한 주 제어기의 사진이다.

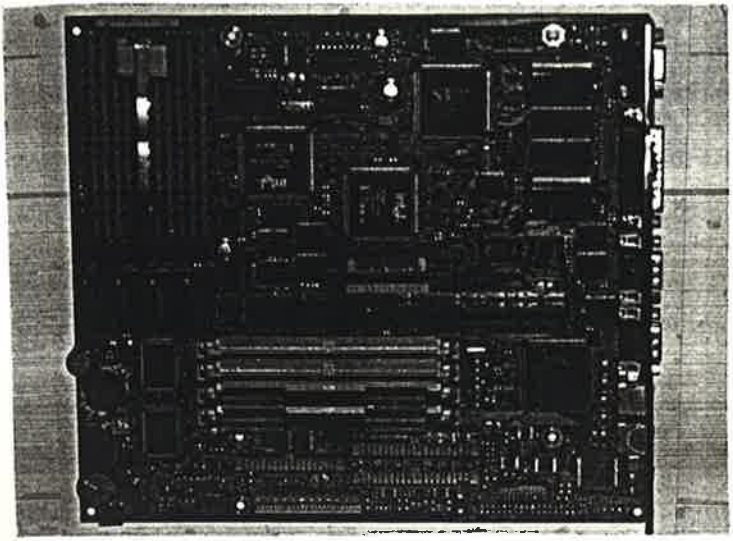


Fig.2-3 Photograph of main controller

2. 모터 제어기의 설계, 제작 및 실험

압출 성형기에서 사용되는 모터는 3상 유도 전동기로 15마력의 main 모터, 1마력의 feeder 모터, 1마력의 cutter 모터가 사용되었다. 그리고 각각의 모터를 제어하기위해서 PWM인버터가 사용이 되었는데 main 모터에는 11[Kw]의 PWM인버터가 사용이 되었고, feeder 모터와 cutter 모터에는 각각 1[Kw]의 인버터를 사용하였다.

Fig.2-4는 인버터와 제어알고리즘을 연결해주는 interface card로 AC모터의 rpm값과 설정값을 전송하는 기능을 한다. 실험에서 사용된 rpm값은 16bit로하여 계측값의 정밀성을 확보하였다.

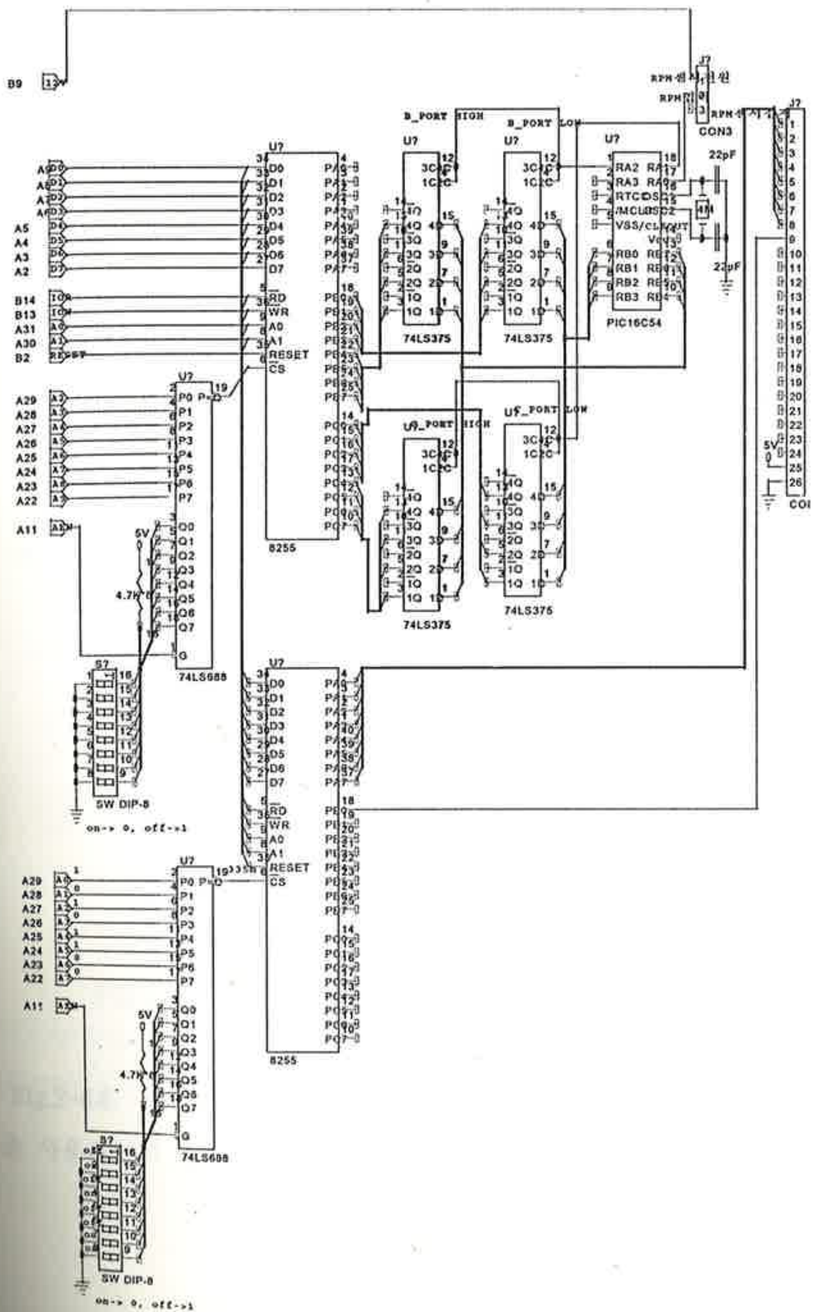


Fig.2-4 Interface Card

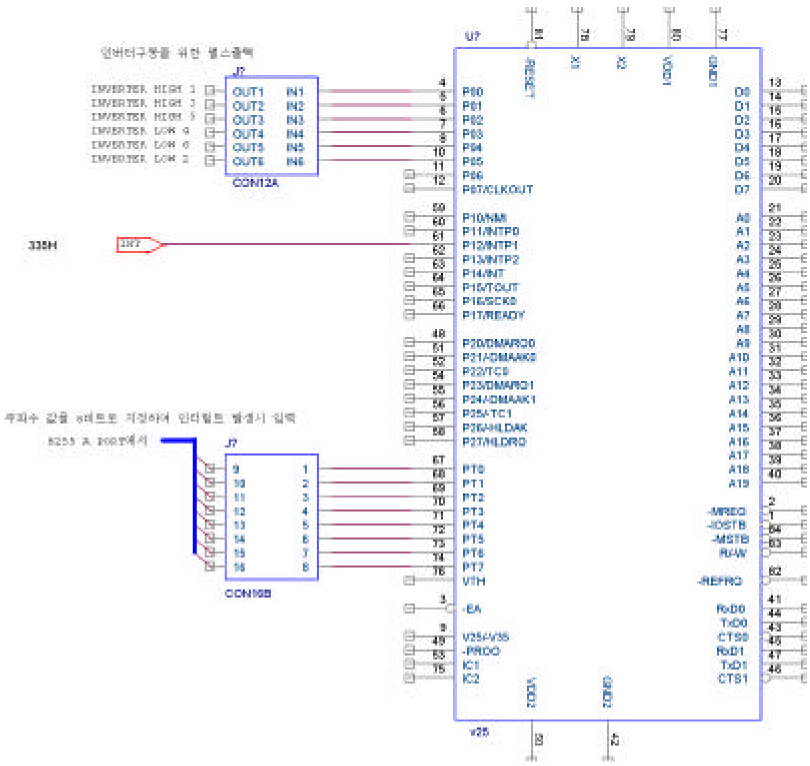


Fig.2-6은 실험제작한 inverter의 회로도이며 스위칭소자로 IR2130을 사용하였고, MOFFET IRF740을 사용하여 제작하였다.

Fig.2-6 Inverter Scheme

PWM인버터를 프로그램에의해 제어하고 통신을 하기위하여 μPD70325마이크로프로세서를 사용하였고, 회로는 Fig.2-7에 있다.

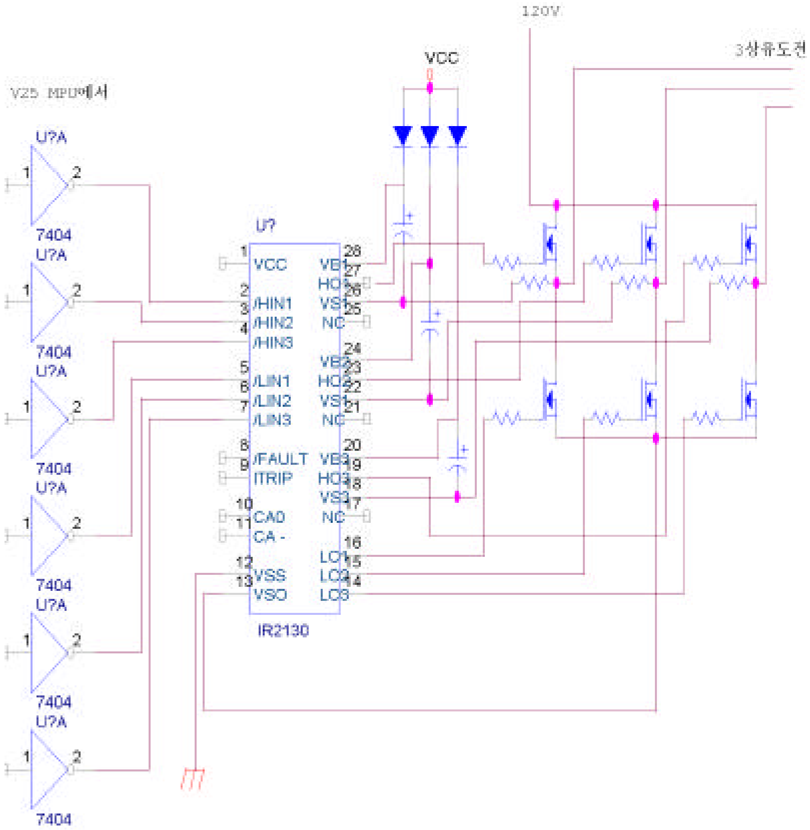


Fig.2-7 PWM inverter controller with a μPD70325.

main motor, feeder motor, cutter motor**에 각각의 컨트롤러가 사용되었으며 컨트롤러의 제어흐름도는** Fig.2-8**과 같다**.

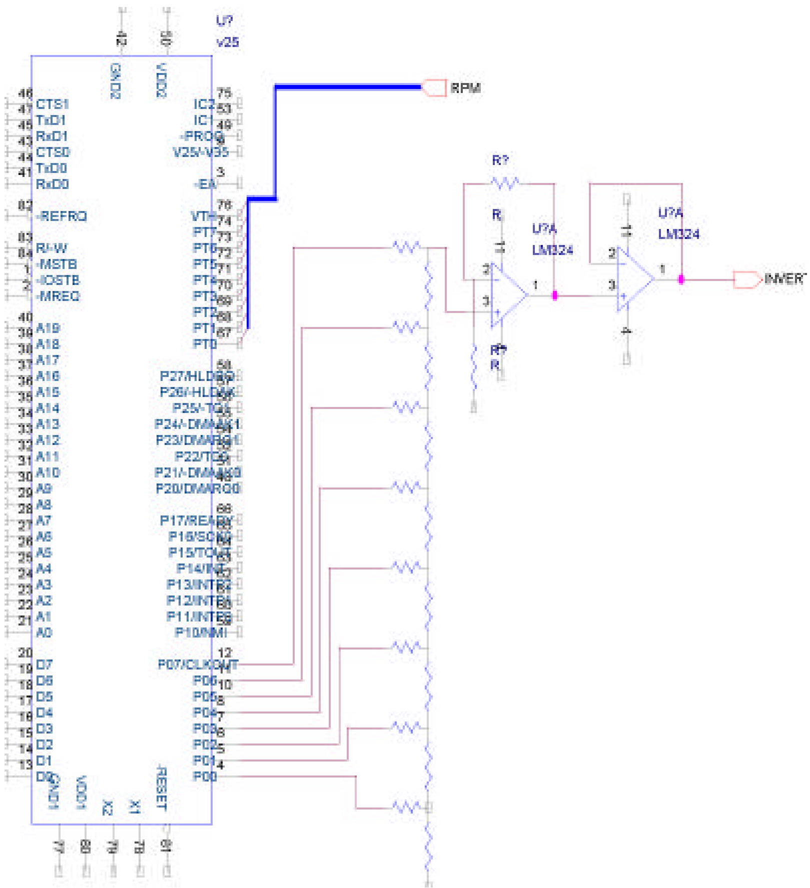
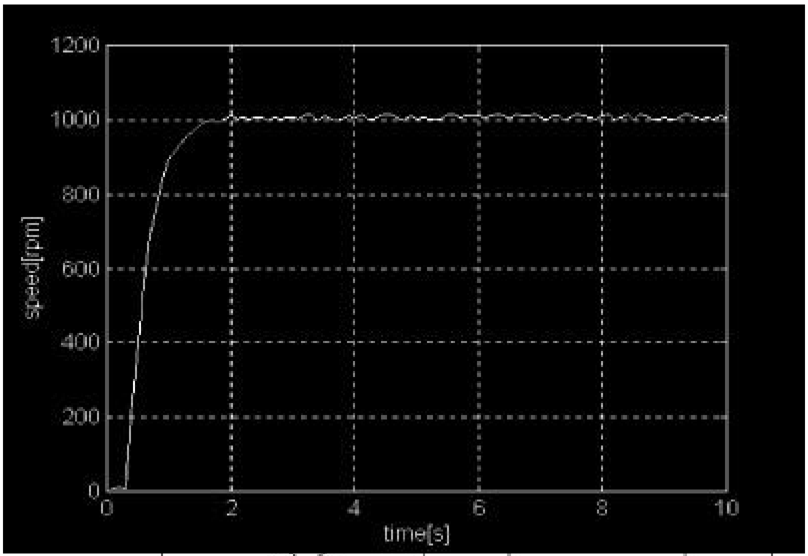
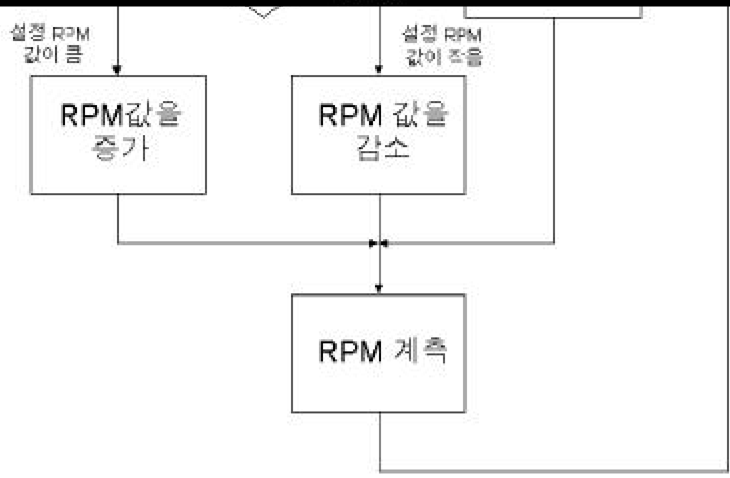


Fig.2-8 Control flowchart of each motor controller
(main, feeder, cutter)

main motor를 대상으로 하여 0 rpm에서 1000 rpm으로 설정하였을 때 motor의 응답곡선은 Fig.2-9과 같다.

Fig.2-9 Response curve of main motor



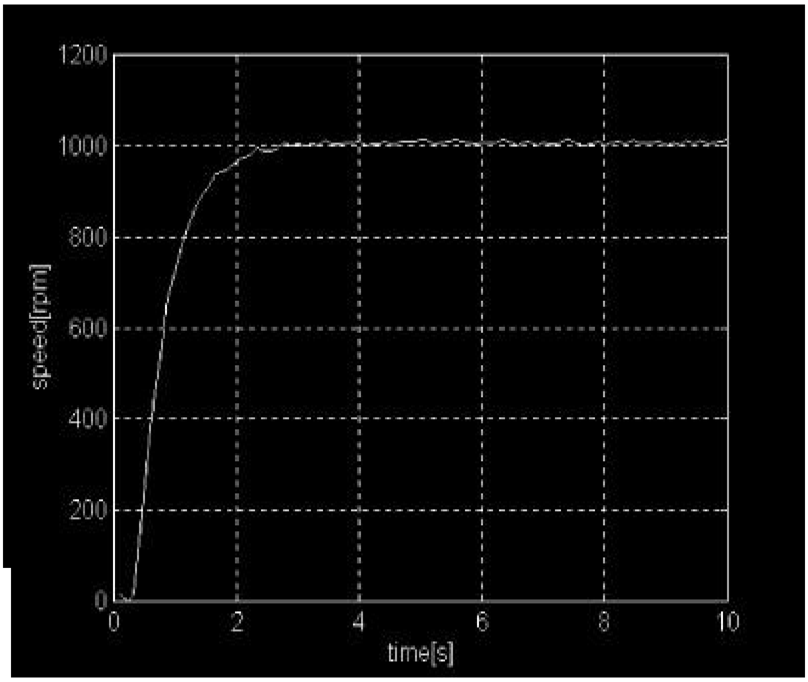


feeder motor를 **대상으로 하여** 0 rpm**에서** 1000 rpm**으로 설정하였을 때** motor**의 응답곡선은** Fig.2-10**와 같다**.

Fig.2-10 Response curve of feeder motor

cutter motor를 **대상으로 하여** 0 rpm**에서** 1000 rpm**으로 설정하였을 때** motor**의 응답곡선은** Fig.2-11**과 같다**.

Fig.2-11 Response of cutter motor



3. **온도 센서 보드의 설계, 제작 및 실험**

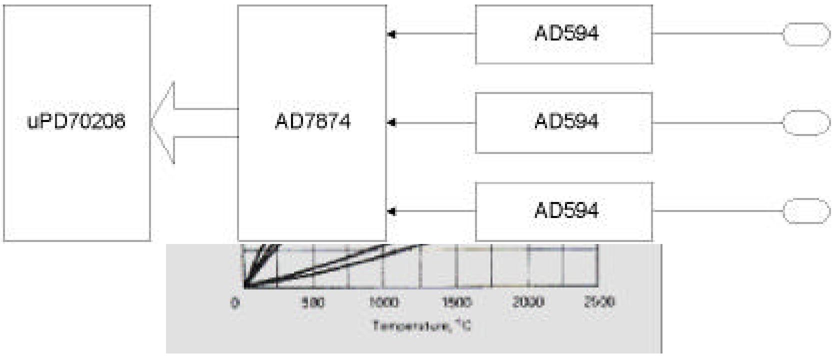
온도 센서 보드는 3**개의 온도센서를 통해 일정시간마다** 3**부분의 온도를 측정한다**.

온도 측정에 사용한 센서는 T **형** thermocouple**로 특성곡선은** Fig.2-12 **이다**.

Fig.2-12 Characteristc curve of thermocouple

온도센서 보드의 블록선도는 Fig.2-13 **이다**.

Fig.2-13 Block diagram of temperature sensor board



온도센서는 Fig.2-12의 특성곡선과 같이 선형적이지 못하므로 선형 보상을 하여야 한다.

선형 보상에는 하드웨어적인 보상과 소프트웨어적인 보상이 있다. 소프트웨어 보상은 온도 측정시간의 감소를 가져오므로 소프트웨어적인 보상은 사용하지 않고 10㎷/℃의 출력을 보장하는 AD594 IC를 사용하는 하드웨어적인 보상회로를 사용하였다. 또한 thermocouple은 접점에의한 온도변화가 생길 수 있으므로 AD594 IC에 최대한 가깝게 연결단자를 설치 하였다. AD594 IC의 10㎷/℃출력은 매우 작은 값이므로 증폭기를 통해 AD변환기로 들어간다. 이때 AD594 IC의 출력은 매우 작아서 전원이나 기타 다른 전기적인 영향에 의해 노이즈가 생긴다. 이러한 노이즈는 하드웨어 필터로서 제거할 수 있으나 전체 크기가 커지고 고정적이어서 마이크로 프로세서 내부에서 디지털 필터를 사용하여 제거한다. AD변환기는 처음에는 8비트의 저속 AD변환기를 사용하였으나 큰 온도영역에서 정밀한 측정을 할 수 없고 변환속도가 느려 온도가 빠르게 변하면 따라갈 수가 없는 단점이 있었다. 그래서 계측 속도와 정밀성을 보완하기 위하여 고속 AD변환기인 AD7874를 사용하여 4채널의 고속 온도측정부를 설계하였다. Fig.2-14은 고속 AD변환기의 회로도 이다.

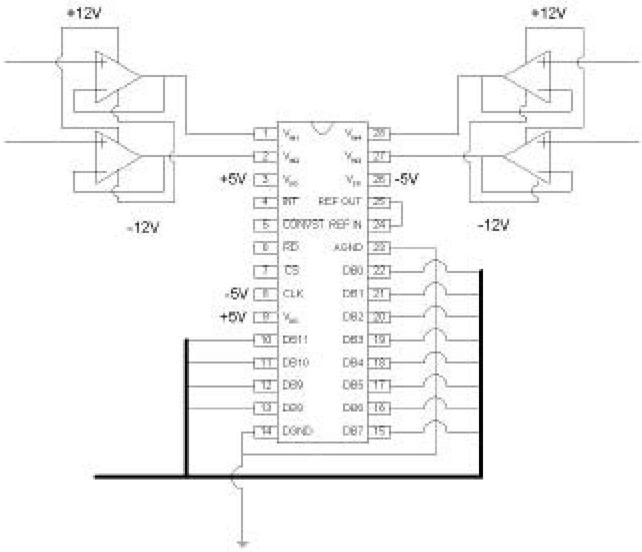


Fig.2-14 Circuit diagram of fast AD converter

Fig.2-14의 변환기는 PD33333 마이크로 프로세서에 연결할때 데이터버스의 비트수가 마이크로 프로세서보다 크다는 문제점이 있다. 이는 다른 어드레스에 버퍼(buffer)역할을 해줄 수 있는 8비트의 래치(latch)를 2개 사용하여 해결 할 수 있었다. 그리고 아날로그 입력단의 임피던스가 조금 낮고 전류를 많이 공급해 주어야 하는 단점은 Fig.2-14과 같이 아날로그 입력단 앞에 전압게인은 1이고 전류만 증폭하는 전류증폭기를 연결하여 해결 하였다.

그러나 이와 같이 설계, 제작하고 실험하였을 때, thermocouple을 extruder를 구동시키지 않고 온도를 측정하면 정상적인 온도값이 측정되나, extruder를 구동하면 extruder의 기계적 진동에 따른 thermocouple의 노이즈가 발생하였다.

Fig.2-15과 2-16은 각 상태에 따른 측정값을 나타낸다.

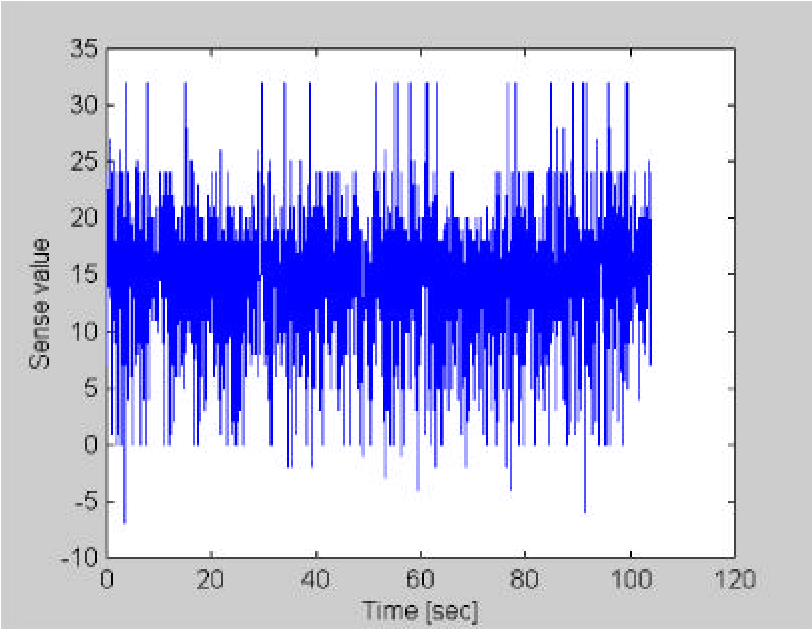


Fig.2- 15　Sensor value without operating extruder

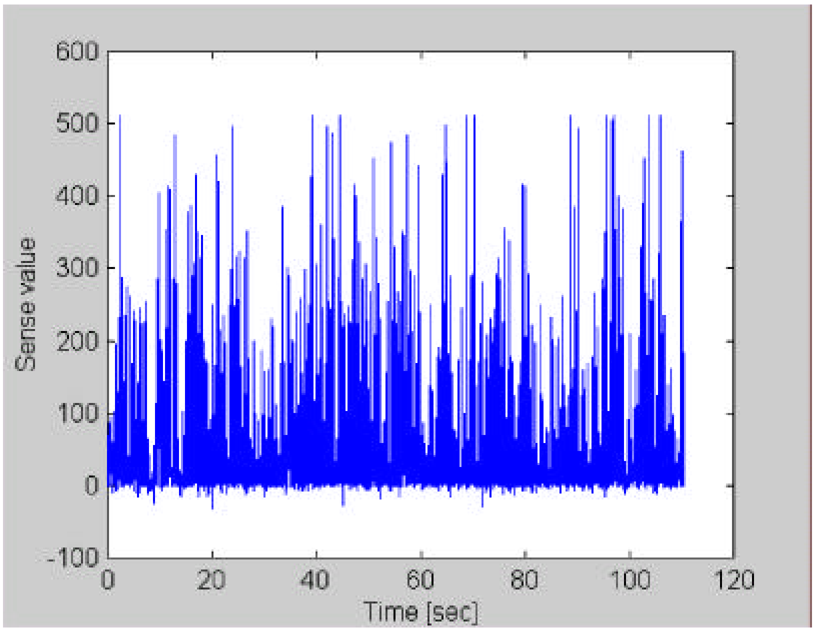


Fig.2-16　Sencor value during operating extruder

Fig.2-15과 2-16에서 알 수 있듯이 압출성형기의 기계적 진동에 따른 thermocouple의 노이즈가 상당히 크게 발생하므로 전처리 과정과 디지털 필터를 설계하여 기계적 진동에 따른 노이즈를 제거하였다. 이때 사용한 디지탈 필터의 특성은 Fig.2-17와 같다.

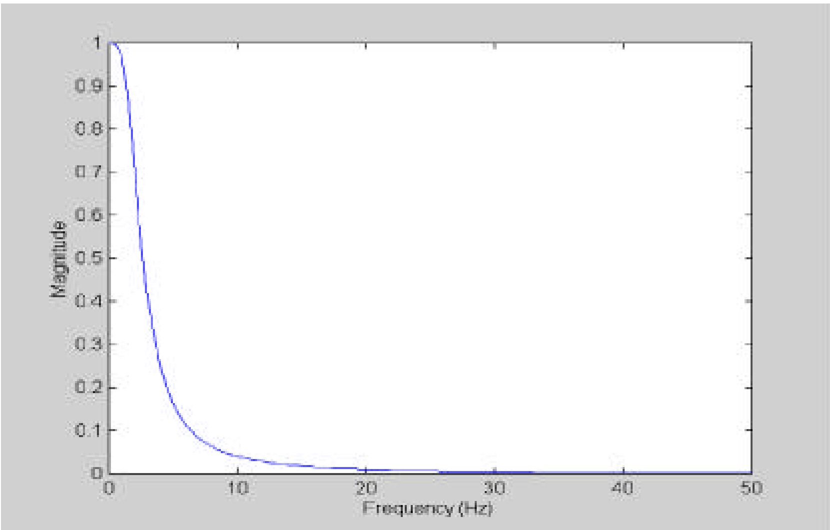


Fig.2-17 Digital filter characteristic

Fig.2-17의 특성을 갖는 필터는 3차 버터워스(Butterworth) 필터로

설계하였고 이때 사용한 계수값은 a=[1 - 1.9643 0.9649], b=[0.1568×10-3 0.3135×10-3 0.1568×10-3] 이다. 그러나 마이크로프로세서의 특성상 정수의 연산만이 가능하고, 16비트의 제한된 워드길이를 가지므로 이와 같은 계수를 그냥 사용하기는 힘이 든다. 그래서 본연구 에서는 입력값과 출력값, 그리고 계수값을 16비트 내의 값을 가지도록 스케일링을 통하여 이를 보정 하였다.

이에 대한 제작 사진은 Fig.2-18 이다.

Fig.2-18 Photograph of temperature sensor board

4. 아날로그 신호를 측정하는 보드의 설계, 제작 및 실험

아날로그 신호를 측정하는 보드는 μPD70208마이크로 프로세서를 사용하여 계측부를 설계 제작 하였으며 이에 대한 제작 사진은 Fig.2-19 이다.

Fig.2-19 Photograph of analog signal sensor board

아날로그 신호를 측정하는 보드는 주로 압력과 파워를 측정한다. 압력계측은 NewYork. LTD.에서 생산한 GP:50 센서를 사용하였다. 이 센서는 1000기압이상의 높은 압력을 0.25%내의 오차를 가지고 측정할 수 있다. 이는 4-20mA의 압력에 비례하는 전류가 흐르므로 이를 전압으로 변화해 주는 장치를 통해 AD변환기를 통해 측정을 한다. 파워는 CHITAI 2402A를 통해 측정한다. 파워는 0-300mV의 파워에 비례하는 전압을 발생시키므로 이를 다시 전압에 비례하는 전류로 전환하여 전송되고 이를 다시 전류에 비례하는 전압으로 바꾸어 AD변환기를 통하여 측정한다. 여기서 전류로 바꾸어 전송시키는 단계는 전압으로 전송할 경우 전송선로에 의한 전압강하가 생기므로 이에의한 측정값에 막대한 오차를 방지하기위하여 사용하였다. 또한 주 제어기와의 통신 오차를 줄이기 위하여 자체 검사 알고리즘을 사용하였다. 통신 오류검사 알고리즘의 흐름도는 Fig.2-20와 같

다.

Fig.2-20 Flowchart of error detect algorithm

유량계측은 CTC CHEM TEC Equipment Company에서 생산한 MAO-250-T-10 AA 센서를 사용한다. 이 센서는 분당 0 500cc를 3%내의 오차를 가지고 측정을 할 수 있다. 이는 0-5V의 압력에 비례하는 전압이 출력되므로 AD변환기를 통해 전압측정을 한다.

5. 유량, 히터, 냉각수 제어 보드의 설계, 제작 및 실험

유량, 히터, 냉각수 제어 보드는 μPD70325 마이크로 컨트롤러를 사용하여 설계 제작하였다. 유량의 측정이나 정확한 조절은 아날로그신호 측정보드와 주 제어기에서 담당하므로 단순한 PAXEG

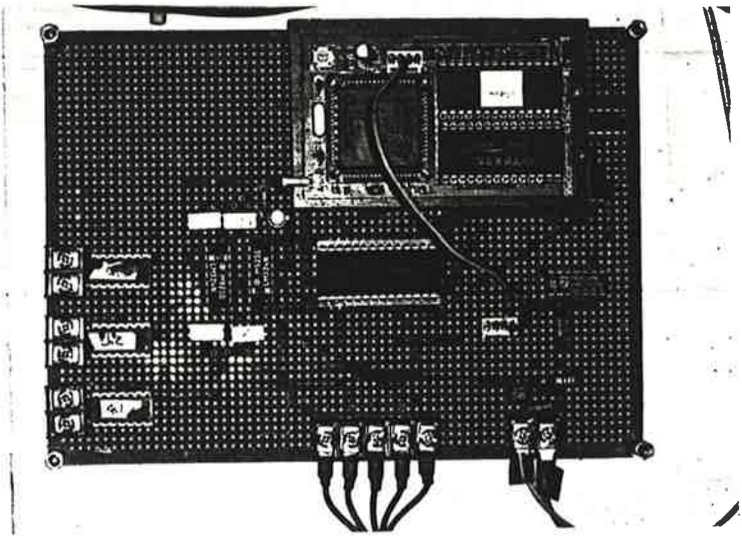


Fig.2-18 Photograph of temperature sensor board

4. 아날로그 신호를 측정하는 보드의 설계, 제작 및 실험

아날로그 신호를 측정하는 보드는 μPD70208마이크로 프로세서를 사용하여 계측부를 설계 제작 하였으며 이에 대한 제작 사진은 Fig.2-19 이다.

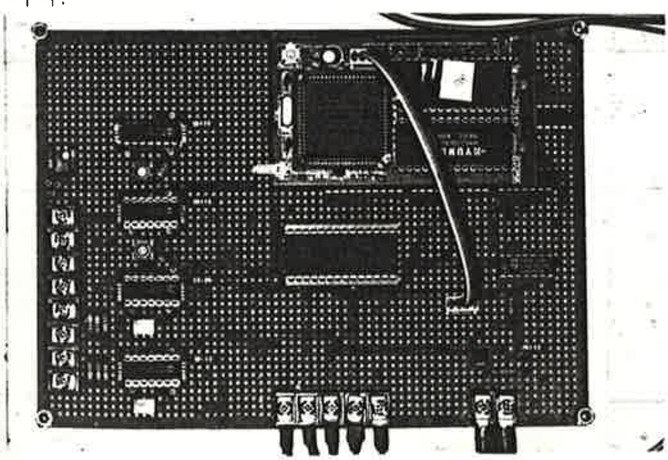
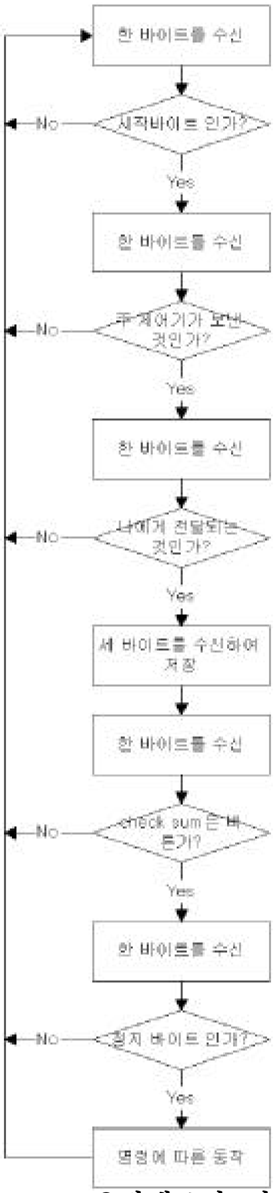


Fig.2-19 Photograph of analog signal sensor board



-205YYP 유량밸브의 연동펌프(peristaltic pump, 0-5V 입력가능한 모델)의 회진수를 제어하도록 0-5 V를 출력하게 설계 제작하였다. 또한 주 제어기와의 통신 오차를 줄이기 위하여 Fig. 2-20의 자체 통신 오류검사 알고리즘을 사용 하였다. 온도의 측정이나 정확한 조절은 온도측정보드와 주 제어기에서 담당하므로 단순한 on-off제어

유량계측은 CTC CHEM TEC Equipment Company에서 생산한 MAO-250-T-10 AA 센서를 사용한다. 이 센서는 분당 0~500cc를 3%내의 오차를 가지고 측정을 할 수 있다. 이는 0-5V의 압력에 비례하는 전압이 출력되므로 AD변환기를 통해 전압측정을 한다.

5. 유량, 히터, 냉각수 제어 보드의 설계, 제작 및 실험

유량, 히터, 냉각수 제어 보드는 μPD70325 마이크로 컨트롤러를 사용하여 설계 제작하였다. 유량의 측정이나 정확한 조절은 아날로그신호 측정보드와 주 제어기에서 담당하므로 단순한 PAXEG-205YYP 유량밸브의 연동펌프(peristaltic pump, 0-5V 입력가능한 모델)의 회진수를 제어하도록 0-5V를 출력하게 설계 제작하였다. 또한 주 제어기와의 통신 오차를 줄이기 위하여 Fig. 2-20의 자체 통신 오류검사 알고리즘을 사용 하였다. 온도의 측정이나 정확한 조절은 온도측정보드와 주 제어기에서 담당하므로 단순한 on-off제어기의 형태로 제작하였다. 그 제작 사진은 Fig.2-21 이다.

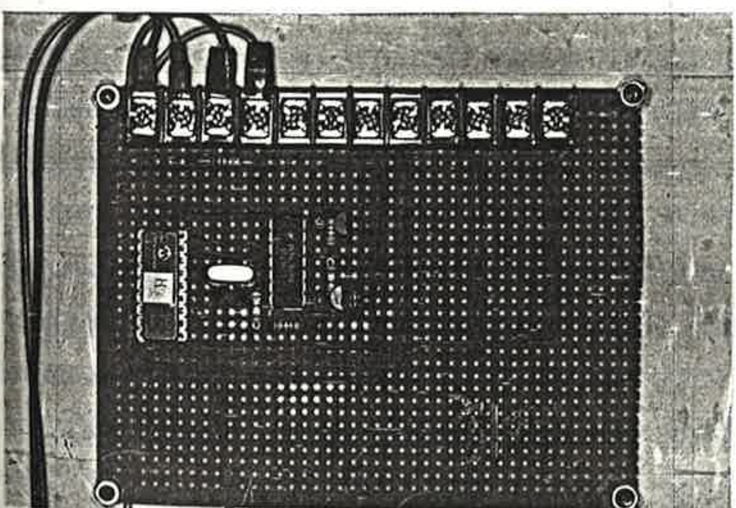


Fig.2-21 Photograph of flow, heater, cooler control board

제 3 장 인공지능제어 소프트웨어의 개발

제1절 서 설

압출성형공정은 1940년대에 사료산업에 처음으로 소개되어 오늘날에는 사료제조에 있어서 필수적인 가공공정으로 각광 받고 있다. 초기에는 주로 애완견사료를 제조하는데 이용되었던 압출성형공정은 각종 양식어류용 양어사료나 소, 돼지, 닭등의 가금류의 사료뿐만 아니라 실험용 동물을 위한 사료를 제조하는데에도 이용되고 있다(1). 또한 식품에 압출성형기술이 이용된 것은 지금으로부터 약 60여년 전이며, 당시에는 주로 마카로니처럼 완전히 익힌 상태가 아닌 반죽상태에 가까운 제품을 느린 속도로 회전하는 스크류에 의해 작은 구멍을 통해 밀어내는 수준이었다. 이처럼 압출성형기술이 이용되기 시작한 이후로 오늘날에 이르기까지 압출성형기의 구조 및 압출성형방식이 관련기술의 발달과 함께 다양해지면서, 질적으로도 거듭 발전을 하게 되었다(2, 3).

압출성형기는 다양한 제품을 생산하는데 유용하게 사용되고 있으며 그 분야가 계속 증가할 전망이다. 하지만, 기능이 다양한만큼 그 내부에서 일어나는 현상도 여러 가지이며, 그 현상에 영향을 미치는 요인들도 매우 다양하고 복잡하기 때문에(4) 어떠한 제품을 생산하는데 있어서 경험적인 방법에 의존하는 경우가 많다. 더구나 사료의 경우에는 제품생산에 사용되는 원료의 종류들이 일반 식품에 비해서 다양하므로 압출성형기 내부에서 일어나는 현상, 원료들이 미치는 영향과 최종제품의 품질과의 어떠한 상관관계를 체계적으로 정립하기가 매우 어렵다. 압출성형공정은 크게 start-up, 정상가동 (steady-operation), shut-down의 3단계로 크게 구분할 수 있는데,

이중에서 우리가 이용하거나 섭취할 수 있는 제품이 나오는 단계인 정상상태를 일정하게 유지하는 것이 제품의 생산 및 품질을 관리하는데 있어서 절대적이다. 정상상태에서는 기계의 작동조건이 잘 맞춰져 있다고 할지라도 원료의 투입량, 원료 자체의 수분함량이나 지방함량, 계절적 요인등에 의한 원료의 품온 및 투입되는 물의 온도의 불균일함등으로 인해 압출물의 품질이 변할 수도 있다. 따라서 이러한 변화를 감지해 낼 수 있는 기준이 필요한데, 일반적으로, 정상상태의 SME(specific mechanical energy), 압출물의 온도, 다이(die) 부근의 압력과 같은 시스템변수(system variables)들이 압출성형공정시 작업자가 직접 조작하는 공정변수에 비해 최종 생산물의 품질과 매우 밀접한 관계가 있는 것으로 알려져 있다(5). 하지만, 지금까지 우리 나라의 압출성형기술을 이용한 제품의 생산과정을 살펴보면, 외국의 설비와 배합 비를 도입하여 간단한 작업자의 교육만으로 압출성형기를 가동하기 때문에, 시스템변수의 조절보다는 공정변수를 맞추는 것에 급급하였다. 따라서 제품의 제조공정을 효율적으로 조절하고 제품의 품질을 유지하는데 있어서 전문 연구인들의 의견보다는 체계화되지 않은 작업자들의 경험이 중요 변수로 작용하게 되었고,. 그 결과 숙련된 작업자에 대한 의존도가 지나치게 높아지고, 제품의 생산 및 품질관리, 신제품의 연구 · 개발에 적지 않은 어려움이 있는 것이 현실이다.

Start-up에 관한 연구로서 Wiedmann(6)에 의하면 압출성형 공정을 인위적으로 조건을 변화시키면서 조심스럽게 시작할 경우에는 원하는 압출성형물을 얻기까지 약 15분이 소요된다고 하였다. 또한 압출성형기를 지나치게 빠르게 start-up 시키면 아직 충분한 열을 받지 못한 차가운 제품이 압출성형기의 plastifying zone으로 유입되어 결과적으로 한계 토크(torque)값을 넘어서 막힘(blockage) 현상이 나타난다. 따라서, 시간과 원료를 절약하고 기계에 무리를 주지 않고

start-up 하기 위해서는, 메인스크류의 구동축에 걸리는 토크값을 계속 측정하면서 한계 토크값 이하의 조건에서 압출성형 공정의 공정변수들을 빠른 시간 내에 증가 시키면서, 정해진 값에 이르도록 제어할 경우에는 그 소요시간이 2분 정도로 단축된다. Bouvier등(7)은 IBM PC와 호환성이 있는 마이크로프로세서 제어시스템을 개발하여 변수들을 제어하면서 그 효과를 연구하였으며, 정상상태에 이르기까지의 과정이 제품의 품질에 영향을 미친다고 하였다. 정상가동이나 shut-down에 관한 연구 보고는 거의 전무한 실정이다.

한편, 여러 분야에서 기계의 자동화로 생산 능력의 향상, 품질 개선, 노동력 절감등의 효과를 보고 있는데, 이러한 자동화 수준도 단순히 마이크로 컴퓨터를 이용한 설정치(set point)의 입력이나 각종 센서로부터의 자료의 수집 등의 단순한 수준을 넘어서 전문가 시스템(expert system)이나 퍼지법칙(fuzzy rules)을 이용한 인공지능제어나 신경망회로(neural net)를 이용한 제어 등으로 그 차원이 높아지고 있다. 압출성형기술을 이용하여 제품을 생산하는 공정도 이러한 첨단 제어기술과 숙달된 작업자의 경험, 연구인들의 이론, 압출성형기의 작동원리등을 종합하고, 컴퓨터 프로그래밍언어를 사용하여 체계적으로 프로그래밍 하여 압출성형기를 자동화한다면, 압출성형기의 활용 및 가동 조건에 대한 이해 부족 등으로 인해 발생할 수 있는 문제점들을 충분히 극복할 수 있을 것이다.

따라서, 본 연구에서는 압출성형기의 가동 중 공정변수나 시스템변수를 계측제어할 수 있는 하드웨어 시스템을 사용하여, 사료생산시 압출성형기를 자동으로 start-up하고, 정상가동을 유지할 수 있고, 자동으로 shut-down시킬 수 있는 소프트웨어를 개발하고자 하였다. 먼저, 가능한 한 포괄적이면서도 다양한 공정조건범위에서 압출성형기를 가동시켜 각각의 공정조건에서 나타나는 공정변수, 시스템변수의 변화와 최종제품의 물성을 측정하여 서로간의 상관관계를 결정하여

전문가 지식을 구축하였다. 이를 바탕으로 퍼지이론과 전문가 지식을 활용하여 제어 알고리즘을 구축하였고, 윈도우용 응용 프로그램 개발 언어인 Visual C++를 이용하여 초보자도 쉽게 다룰 수 있는 압출 성형기 가동용 소프트웨어를 완성하여, 그 결과 제품의 생산 과정 중에 어떠한 요인의 변화에도 압출성형기 스스로 그 변화에 적응할 수 있는 능력을 갖는 인공지능제어 시스템을 완성하였다.

제2절 애완견사료 압출성형 전문가 지식의 구축

1. Start-up

가. 재료 및 방법

본 실험에서 사용된 압출성형기(FESTINA-40, (주)남성)는 쌍축형(twin) 압출성형기로서 원료투입구, 배럴(barrel), 스크류(screw) 및 다이(die)로 구성되어 있다. 스크류의 직경은 43㎜, 배럴길이 700㎜이고 스크류는 동방향 자가청소형으로서 L/D는 16이고 다이쪽으로 갈수록 pitch가 감소하는 조합이며 feeder는 용적 식으로 단축스크류가 장착된 원료투입기를 이용하였다. 히터는 2단으로 장착되어 있다.

애완견사료를 대상으로 본 실험에서 사용된 원료는 (주)신촌사료에서 생산된 배합원료로서 그 배합 비는 Table 3-1과 같다.

전문가 지식 구축을 위한 실험으로 start-up의 공정변수는 Table 3-2에 의하여 정하였고 이의 순서에 따라 start-up을 시작하였는데, 우선 기기를 작동하기 전에 미리 히터를 on상태에 놓아 배럴의 온도가 120℃에 도달하였을 때 기기 작동을 시작하였다. 먼저 condition 1에 의하여 수분공급량을 급속히 감소시키고 아울러 원료공급속도를 급격히 증가하도록 설계하여 실험을 하였으며 또한 condition 2에 의

하여 condition 1보다는 수분공급량을 완만히 감소시켰고 원료공급속도는 완만히 증가하도록 설계하였으며 condition 1보다는 높은 온도의 상태에서 실험을 하였다. 여기서 공정변수의 선정기준은 많은 실험을 통한 경험에 의하여 산출하였다.

나. 결과 및 고찰

Table 3-2의 조건에서 수동으로 start-up의 실험을 한 결과, 약 30초 내에 공정변수가 목표치에 도달하였을 때 구동모터 power는 크게 증가하지 않았고 압력은 급격히 증가하여 곧 일정하게 유지하였다. 그러나 다이에서 압출물의 온도는 느리게 증가하여 일정한 값에 수렴하는데 장시간이 소요되었다 (Fig. 3-1). 그런데 다이의 온도가 더 높은 상태에서 수동으로 실험을 한 결과, 다이에서 압출물의 온도가 일정한 값에 수렴하는데 걸리는 시간이 단축되었다 (Fig. 3-2). 따라서 start-up은 다이직전의 압출물 온도를 되도록 빨리 일정한 값에 수렴시키는 것에 제어 알고리즘의 주요 과제로 삼아야 한다는 것을 알 수 있었다. 또한 power의 측정치를 비교해 보면 높은 상태에서 시작했을 경우가 낮은 상태에서 시작했을 경우보다 낮음을 알 수가 있었다. 이는 다이온도가 높은 것의 열에너지가 원료에까지 기계적 에너지로 전달받게 되어 원료는 열전달을 받아 그만큼 점성이 약하여졌기 때문에 스크류축에 걸리는 torque값이 낮기 때문일 것이라고 생각된다.

2. 정상 가동

가. 재료 및 방법

(1) 압출성형

프로그램 개발에 필요한 전문가 지식을 얻기 위해 가능한 한 포괄적이면서도 다양한 공정조건 범위에서 압출성형기를 가동시켜 각각의 공정조건에서 나타나는 시스템변수의 변화와 최종제품의 물성을 측정하였다. 실험은 3번 반복실험하였다.

Table 3-3에 나타난 공정조건에 맞추어 실험을 하였으며, 처음 압출성형기를 가동할 경우에는 start-up하여 시스템변수가 일정하게 유지된 후에 압출물을 취하여 분석 시료로 사용하였다. 또한 각 공정조건이 변할 때마다 충분한 시간 간격을 두어 이전의 공정조건의 영향을 최소화 하였다.

(2) 압출물의 물성측정

(가) 팽화율 측정

팽화율은 다이 구멍의 직경에 대한 압출물의 직경비로 정의되며, 각각의 공정조건에 대해서 압출물을 30 ㎝ 길이로 취하여 임의로 10 부위를 택하여 버어니어캘리퍼스로 직경을 측정하고, 다음 식에 의해 구한 후 그 평균값을 산출하였다(8).

$$\text{팽화율} = \frac{\text{압출성형물의 직경}}{\text{Die의 직경}}$$

(나) 집단밀도 측정

집단밀도는 제품을 포장, 저장 및 수송을 계획하는데 있어서 필수적인 물리량으로서 부피를 알고 있는 용기에 제품을 채운 후 그 질량을 측정하여 구한다. 이 경우 충진하는 방법에 따라 그 값이 달라지므로 충진방법을 동일하게 적용하고 반드시 명시한다(9). 본 실험에서는 500 ml 메스실린더에 건조시킨 압출성형물을 천천히 부으면서 부피의 변화가 없을때까지 가볍게 메스실린더 밑부분을 쳐주었으며 500 ml가 채워졌을 때의 무게를 측정하여 g/ml단위로 측정하였다(10).

(다) 수분흡수도 측정

수분흡수도는 건조공정을 거친 압출성형물 100 g을 2개의 sieve 사이에 넣고 25℃ 물에 잠기도록한 후에 5 분 동안 방치한 다음, 꺼내어 2 분 동안 방치하면서 drain시킨 후 무게를 측정하고 다음 식에 의하여 측정하였다(11).

$$\text{수분흡수도} = \frac{\text{수화시킨 압출물(g)} - \text{건조압출물(g)}}{\text{건조압출물(g)}}$$

(라) 통계처리

공정변수및 시스템변수의 압출물의 물성에 대한 관계를 SAS통계 패키지의 상관계수분석법을 이용하여 상관계수로 나타내었으며, 공정변수에 따른 시스템변수의 변화를 RSM을 이용하여 그래프로 나타내었다.

나. 결과 및 고찰

압출물의 물성 측정치를 Table 3-4에 나타내었으며, 그에 따른 공정변수, 시스템변수 및 압출물의 물성간의 상관관계를 나타내는 상관계수를 Table 3-5에 나타내었다. 우선 공정변수의 변화에 따른 시스템변수의 변화를 살펴보면, 배럴의 온도가 상승함에 따라, 시스템변수인 SME와 다이부근의 압력은 감소하였으며, 압출물의 온도는 상승하였다. 이것은 배럴을 감싸고 있는 히터를 통해 전달되는 열에너지가 증가함에 따라 상대적으로 기계적에너지 투입이 감소되는 현상이며, 내부 물질의 온도 상승에 의해 점도가 낮아지면서 압력이 떨어지게 된 것이다. 한편 투입되는 물의 양이 증가할수록, SME, 다이

부근의 압력, 압출물의 온도 모두 감소하는 경향을 보였다. 이것은 투입되는 물의 양이 증가하면서, 내부 물질의 온도와 점도를 떨어뜨리게 되고, 그에 따라 스크류의 회전이 비교적 원할하게 되면서 스크류의 회전에 의해 내부물질로 전달되는 기계적에너지의 영향이 감소하여 일어나는 현상으로 사료된다. 또한 메인스크류의 회전속도가 증가함에 따라, SME와 압출물의 온도는 상승하였으며, 다이부근의 압력은 감소하였다. 이것은 메인스크류의 회전속도 증가에 따라 기계적에너지 투입이 증가되면서 내부 물질의 온도가 상승하며 그에 따라 점도가 변화하면서 압력이 떨어진 것이다(6).

압출물의 물성은 배럴의 온도가 증가함에 따라 팽화율 및 흡수도는 증가하였으며, 집단밀도는 감소하였다. 투입되는 물의 양이 증가할수록 팽화율 및 흡수도는 감소하였고, 집단밀도는 증가하였다. 메인스크류의 회전속도의 증가에 따라 팽화율 및 흡수도는 증가하였고, 집단밀도는 감소하였다. 이것은 Meuser 와 Wiedmann(6)의 결과와 같다.

공정변수와 시스템변수 중에서 압출물의 물성에 영향을 미치는 정도가 강한 것을 알아보기 위해 구한 상관계수의 값을 살펴보면, 압출물의 물성에 가장 크게 영향을 미치는 것은 투입되는 물의 양으로 나타났지만, 전반적으로 공정변수에 비해서 시스템변수가 압출물의 물성과 상관관계가 더 있는 것으로 나타났다. 즉, 수분 함량은 팽화율과 -0.8734로 강한 음의 상관관계를 보였으며, 집단밀도에 대해서는 0.8799로 양의 상관관계를, 흡수도에 대해서는 -0.8654로 음의 상관관계로서 압출물의 물성에 대해 가장 상관관계가 높게 나타났다. 이것으로부터 압출물의 물성 및 그 물성을 형성하기 위해 지표가 되는 시스템변수에 수분함량이 가장 큰 영향을 미친다는 사실을 알 수 있었다. 하지만, 배럴 온도와 메인스크류의 회전속도의 압출물의 물성에 대한 상관관계는 그 계수가 ±0.4000이하의 값으로 낮게 나타났

으며, 반면 SME, 압출물의 온도 및 다이부근의 압력과 같은 시스템 변수는 압출물의 물성에 대한 상관계수가 ±0.6000이상으로 공정변수의 것에 비해 높게 나타났다. 이 사실로부터 압출물의 물성의 변화에 좀 더 밀접한 관계가 있는 것이 시스템변수임을 확인하였다. 이러한 사실은 Wiedmann 과 Strecker(5)에 의해서도 보고된 바 있다.

공정변수에 비해 압출물의 물성과 상관관계가 높게 나타난 시스템변수를 조절하기 위해, 공정변수가 시스템변수에 영향을 미치는 경향을 Table 3-4에서 나타낸 것보다 자세하게 파악하고자, 비교적 시스템변수의 변화에 영향을 미치는 정도가 적은 배럴의 온도를 고정변수로 하고, 수분 함량(15 25%)과 스크류의 속도(120 180 rpm)가 주어진 범위에서 변할 때, 각각의 시스템변수들이 변화하는 경향을 SAS 통계 패키지를 이용하여 3차원 그래픽 반응 표면으로 나타내었다(Figs. 3-3 3-11). Fig. 3-3은 배럴온도가 60-80℃인 조건하에서 수분함량이 15 25%, 메인스크류의 회전속도가 120 180 rpm 범위에서 변할 때, SME의 변화 경향을 나타낸 것이다. SME는 수분함량이 높아질수록 감소하는 경향을 보였으며, 메인스크류의 속도가 증가할수록 증가하는 경향을 나타내었다. Fig. 3-4는 Fig. 3-3과 같은 조건에서 다이부근의 압력의 변화 경향을 나타낸 결과이다. 수분함량이 증가할수록 압력은 떨어지고, 메인스크류의 회전속도가 증가할수록 감소하는 경향을 볼 수 있다. Fig. 3-5는 압출물의 온도가 변화하는 경향을 나타낸 것으로 수분함량이 증가할수록 압출물의 온도는 떨어지는 경향이 있었으며, 메인스크류의 회전속도가 증가할수록 같이 상승하는 것을 볼 수 있다. 나머지 배럴 온도 조건인 100-120℃, 140-160℃의 경우에도 이와 비슷한 경향을 나타내었다. 따라서, Tables 3-4, 3-5와 Figs. 3-3 3-11 의 결과로부터 뒤에서 언급될 시스템변수를 조절하기 위한 퍼지함수와 전문가지식을 구축하게 되었다.

3. Shut-down

가. 재료 및 방법

전문가 지식 구축을 위한 실험으로 shut-down의 공정변수는 Table 3-6에 의하여 정하였고 이의 순서에 따라 shut-down을 시작하였는데, 우선 shut-down이 시작된 지 30초 후에 구동모터 회전속도는 150rpm으로 줄였는데 이는 150rpm보다 낮게 감소시켰을 경우 원료를 사출구 밖으로 밀어내는 힘의 부족으로 인하여 막힘현상이 일어났기 때문에 최소한의 힘으로 설정하였으며 feeder rpm은 서서히 감소시켰고 수분공급속도는 condition 1에서는 56rpm까지, condition 2에서는 64rpm까지 올려 shut-down을 실시하였다. 여기서 공정변수의 선정기준은 많은 실험을 통한 경험에 의하여 산출하였다.

나. 결과 및 고찰

Shut-down의 제어시 가장 주의해야 할 사항은 압출성형기내의 원료(충전물)가 남아 있는데도 불구하고 더 이상 배출이 안될 경우, 즉 막힘현상이다. 막힘현상은 공정변수를 정지상태로 서서히 줄이는 과정에서 배럴내 충전물을 밀어내는 힘이 부족하게 되어 더 이상의 사출은 없고 스크류만 겉돌면서 일어난다. 수동으로 shut-down을 한 결과 구동모터 회전속도를 너무 빨리 감소시키거나 물공급속도를 너무 급격히 증가시켰을 경우, 또는 원료공급속도를 너무 급격히 감소시켰을 때 막힘현상이 일어남을 알 수가 있었다. 구동모터회전속도를 고정한다고 가정하면, 물 또는 원료공급속도에 의해 배럴내 충전물의 수분함량이 결정되어 수분함량을 너무 급격히 높게 조절하면 스크류가 회전할 지라도 충전물은 사출되지 않고 스크류는 겉돌며 막히게 된다. 또한 데이터로 나타낼 수는 없었지만 막힘현상이 일어났을 경우, 구동모터의 power는 기본에너지에 도달하기 이전에 감소하는 일

없이 일정함을 유지하거나 상승하는 현상이 관찰되었고 반면에, 막힘 현상이 없이 매끄럽게 shut-down이 완료되었을 경우의 구동모터 power는 기본에너지까지 상승하거나 일정하게 유지되는 일없이 부드럽게 감소하는 것을 볼 수가 있었다 (Figs. 3-12, 3-13). 이와 같은 사실로 미루어 보아, 배럴내 충전물을 밀어내는 힘의 상태를 나타내는 척도로서 구동모터의 power를 간주하여 막힘현상이 일어나지 않도록 공정변수를 제어하도록 알고리즘이 작성되어야 함을 알 수가 있었다.

제3절 소프트웨어의 구성

1. Visual C++ 프로그래밍

컴퓨터가 작동하는데 기본이 되는 OS의 대부분이 윈도우즈 환경을 지원하는 최근 여러 프로그램들의 경향을 살펴보면 기존의 DOS용 프로그램보다는 윈도우즈용 프로그램이 대다수를 차지하고 있다.

윈도우즈용 응용 프로그램의 특징이라함은 객체 지향적 프로그램이며 이벤트(event) 중심의 프로그램을 들 수 있다. 즉 프로그램이 진행되는 과정이 어떤 프로그래밍된 순서에 의해 진행되는 것이 아니라, 사용자에 의해 선택된 명령에 해당되는 부분이 실행된다. 그리고 프로그램을 실행하고 사용하는데 기존의 도스용 프로그램이 정해진 명령어를 기억하여 사용자가 직접 입력하거나 단축키를 이용하여야만 했던 것과는 달리, 화면(윈도우즈)상에 보이는 메뉴나 툴바를 마우스를 이용하여 단순히 '클릭' 기능을 이용한다. 또한 도스용 프로그램은 프로그램마다 개발자들이 고유로 지정한 단축키들이 다르기 때문에 프로그램마다 반드시 매뉴얼이 필요했다. 하지만, 윈도우즈 응용 프로그램들은 그 기본이 되는 틀 즉, 사용자 인터페이스 메커니

즘이 비슷하므로 몇 번 정도 윈도우즈 응용 프로그램을 실행해 본 경험이 있는 사람들이라면, 설명서 없이도 프로그램을 다룰 수 있는 장점이 있다.

일반 제어용 프로그램들도 이러한 추세에 맞추어 점차 도스 환경용 프로그램에서 윈도우즈용으로 변화하고 있다. 사용자들은 화면에 나타나 있는 버튼, 메뉴로부터 명령을 쉽게 선택함으로써 제어 대상으로부터 값을 받아들이기도 하고 그 값에 대응하는 제어 명령을 보내기도 한다. 또한 그 수행되는 과정을 도스 환경에서 보다 훨씬 다양하면서도 정교한 화면으로 쉽게 확인할 수 있게 되었다.

이러한 윈도우즈 응용 프로그램을 제작하는데 이용되는 툴(tool)에는 Visual Basic, Power Builder, Delphi등 여러 가지가 있으나, 본 프로젝트에서 사용한 것은 Visual C++이다. Visual C++의 경우 프로그래밍 과정이 다른 것들에 비해 복잡한 경향이 있으나, 프로그래밍에 사용되는 언어의 기본이 기존의 C 언어나 C++ 언어이므로 다른 개발 툴에 비해 하드웨어 제어용 프로그램을 개발하는데 용이하다.

Visual C++을 이용하여 제어용 프로그램을 개발하는 과정은 크게 프로그램의 틀과 화면을 구성하는 '디자인(design)' 과정과 그 화면 구성 요소와 필요한 기능을 수행할 수 있도록 프로그래머가 직접 프로그램 소스를 입력하는 '코딩(coding)' 과정으로 구분할 수 있다. 디자인 과정에서는 다이얼로그 박스, 명령버튼, 에디트상자, 그래프환경, 그리드(grid)상자 등 제공되는 여러 가지 도구들을 이용하여 프로그램의 화면을 구성한다. 그 다음 코딩 과정에서 C/C++언어나 Visual C++에서 지원되는 명령을 이용하여 프로그램 소스를 입력한다.

Fig. 3-14는 Visual C++을 이용하여 압출성형기를 제어하는 프로그램의 주요 루프를 나타낸 그림이다. 그림에서 CMainFrame, CComm등 'C ' 의 형식의 그룹은 프로그램을 구성하는 클래스들의

이름을 뜻하며, 각각 역할이 다르다. CMainFrame의 경우에는 프로그램의 가장 기본이 되는 틀을 구성하는 역할을 하며, CAutoView는 사용자가 실제로 볼 수 있는 화면을 처리하는데 이용되는 함수 및 변수들의 집합이며, CComm은 본 프로그램에서 가장 중요한 역할을 하는 데이터 통신에 관한 일을 하는 클래스이다.

프로그램은 사용자가 화면에 나타난 'AUTOMATIC' 이나 'MANUAL' 이란 버튼을 눌렀을 때, 실행되는 OnAutomaticOp() 또는 OnManualOp()가 실행되면서 시작된다. 이에 의해 실제 데이터 통신이 진행되는 CMainFrame에서 StartCom()이라는 함수에 의해 통신 보조 클래스인 CComm의 SetComPort(), OpenComPort()를 실행시켜 통신환경을 설정하고, 포트를 열어 놓는다. 한편, CMainFrame의 StartCom()이 실행되면서 설정해 놓은 타이머에 의해 SendToReceive()라는 함수에 의해 마이크로프로세서에 명령이 전달되면, 마이크로프로세서에서 해당되는 데이터를 보내며, 이것을 OnReceiveData()가 받아들이면서 버퍼에 저장해 놓으면 GetData()에 의해 하나씩 버퍼에서 꺼내면서 고유 데이터블록 구조와 확인을 하여 실제 데이터만을 선택한다. 이러한 실제 데이터들은 사용자가 볼 수 있도록 화면에 나타나기도 하고 또한 다음 제어과정을 위해 퍼지 추론의 기본 데이터로 이용되기도 한다.

지금까지 설명한 과정을 반복하면서 압출성형공정은 제어되며, 자세한 내용은 '2. Graphic user interface의 구성', '3. 데이터 통신 방식(RS-485통신)', '4. 데이터 입력과 출력' 에 설명하였다.

2. Graphic user interface의 구성

가. AppWizard를 이용한 프로그램의 기초 구성

AppWizard는 말 그대로 제작하고자 하는 프로그램의 기본 틀을 구성하는데 '마법사'와 같은 역할을 하는 Visual C++에서 제공되는 기능으로서, 프로그래머는 자신의 계획에 맞추어 AppWizard의 과정을 거치면서 프로그램의 틀을 구성한다. 이 과정에서 프로그래머들은 어떠한 코딩작업을 하지 않아도 하나의 윈도우즈 프로그램이 만들어지게 된다.

본 프로그램에서는 AppWizard를 통해 하나의 화면에 하나의 도큐먼트(document)를 갖는 SDI(single document interface) 형식을 취하였으며, 데이터베이스 지원은 받지 않으며, OLE(object linking and embedding) 지원은 받으며, 화면구성에 있어서는 툴바(toolbar), 스테이터스바(status bar), 인쇄 및 미리보기 기능은 기본으로 설정하였다. 특히 화면에 나타나는 뷰(view)부분을 CFormView형태를 이용하여 마치 다이얼로그 박스를 사용하듯 화면에 버튼, 에디트상자, 그래프환경, 그리드상자 등의 도구를 이용할 수 있도록 하였다.

나. Resource editor에 의한 화면 구성

화면 구성은 resource editor를 이용하여 직접 구성하였으며, 각각의 화면들에는 버튼형태의 명령 입력 장치를 나타내었다. 초기환경에는 압출성형기를 mspaint를 이용하여 표현한 비트맵 파일과 명령버튼, 에디트상자등을 이용하여 구성하였고, 그래프환경은 Visual Basic에서 지원되는 pinnacle-graph control (graph32.ocx)과 명령버튼을 사용하였으며, 그리드(grid)환경은 Visual C++에서 제공되는, grid control (grid32.ocx)과 에디트상자, 명령버튼을 사용하였다.

다. Class Wizard를 이용한 편리한 프로그래밍

프로그램 내에서 사용되는 명령어, 함수 및 변수들을 연결하거나, 새로운 함수를 만들고, 새로운 변수를 지정하는데 시간을 절약하고 혼동되지 않도록, Class Wizard를 이용하였다.

라. 프로그램 진행

압출성형기 자동제어용 프로그램을 실행시키면, Fig. 3-15과 같은 초기화면이 나타난다. 초기화면은 모두 8개의 명령버튼으로 구성되며, 그 명령들 및 부수적인 명령들을 각각 메뉴바와 툴바에 나타내었다. 우선 사용자가 '통신환경설정' 버튼을 누르면 통신포트와 통신속도를 결정하는 다이얼로그 박스(Fig. 3-16)가 나타난다. 적당한 통신포트와 속도를 선택하고 'OK' 버튼을 누르면, 통신환경설정이 완료된다. 그 다음 원하는 공정조건을 설정하기 위해 초기화면에 나타낸 압출성형기의 주변에 있는 SP(set point) 에디트상자에 값들을 입력한다. 하나의 에디트상자에 값을 입력한 다음에 다음으로 이동할 경우에는 키보드의 'Tab' 키를 이용하거나 마우스를 이용한다. 모든 공정변수를 입력한 다음 사용자는 '공정변수설정' 버튼을 누르면 공정변수입력이 완료된다. 지금까지의 과정은 압출성형기를 자동제어하는데 있어서의 기본환경 및 조건을 설정하는 과정이며, 아직까지는 컴퓨터와 제어용 하드웨어간의 통신은 진행되지 않는 상태이다.

제어의 시작은 'AUTOMATIC' 버튼이나 'MANUAL' 버튼에 의해서 진행된다. 'AUTOMATIC'의 경우에는 버튼을 누름으로써 압출성형기의 start-up에서부터 정상가동의 과정이 짜여진 제어로직에 의해 진행된다. 즉, 먼저 바렐의 온도를 설정치까지 높이기 위해 히터가 가동되며, 온도가 설정치에 다다르면 메인스크류, 피더(feeder) 스크류의 회전속도 및 수분공급장치의 회전속도가 정해진 시간 내에 처음에 입력한 설정치까지 다다르게 된다. 본 프로그램에서는 여러

번의 실험결과에 의해 30 초 내에 모든 공정조건이 설정치에 도달하도록 하였다. 그 후 측정되는 모터의 전력, 다이압력, 압출물의 온도값에 의해 공정이 정상가동으로 전환된다. 정상가동 상태는 시스템변수의 값을 계속 측정하면서 퍼지추론 과정을 거쳐 그 출력값으로 나오는 공정변수의 값을 변형시키면서 유지하게 된다. 공정 중에 갑작스런 위급상황이 발생하면 경고음, 경광등이 작동하면서 프로그램은 shut-down과정으로 전환된다. 그리고 사용자가 임의로 압출성형공정을 중단하고자 할 경우에는 'SHUTDOWN' 버튼을 누르면 shut-down제어로직에 의해 다이구멍의 막힘 현상없이 모든 공정 조건이 제로가 된다. 'MANUAL'의 경우에는 공정의 start-up과정에 소요되는 시간을 정할 수 있다는 점에서 'AUTOMATIC'과 다르다. 'MANUAL' 버튼을 누르면 소요시간을 설정하는 다이얼로그 박스(Fig. 3-17)가 나타나며, 사용자는 start-up에 소요되는 시간을 임으로 설정할 수 있다. 소요시간을 입력한 후 'OK' 버튼을 누르면 공정이 시작되며 그 후의 과정은 'AUTOMATIC'과 같다. 초기환경에서 'CLEAR' 버튼은 모든 공정변수의 현재값(PV)과 설정치(SV)를 초기화하는 기능을 한다. '그래프조건설정' 버튼을 누르면 그래프환경에 나타낼 공정변수와 시스템변수를 선택하고, 그래프 타이틀을 입력하며, 데이터수집주기를 결정하는 그래프옵션 다이얼로그 박스(Fig. 3-18)가 나타난다. 만일 통신환경이나 공정조건을 설정하지 않은 상태에서 'AUTOMATIC' 버튼이나 'MANUAL' 버튼을 누르면 조건들이 설정되지 않았음을 알리는 메시지박스가 나타나면서 통신환경이나 공정조건을 설정해야함을 알려준다.

초기화면의 메뉴의 '환경선택'에서 그래프환경을 선택하면 화면이 초기화면에서 Fig. 3-19와 같은 그래프환경으로 전환된다. 그래프환경에서 '▶' 버튼을 누르면 수신되는 데이터들의 값에 따라 그래프가 그려지기 시작하며, '■' 버튼을 누르면 중단된다. 그리고, 'ⓟ'

버튼을 누르면 그래프를 인쇄한다. 그래프환경에서 환경선택 메뉴를 이용하여 초기화면이나 뉴메릭(numeric)환경(Fig. 3-20)으로 전환될 수 있다. 뉴메릭환경은 수신되는 데이터를 정해진 각각의 격자(grid)에 나타내는 화면으로 초기환경에서 설정된 공정조건이 에디트상자에 표시되어 있기 때문에 수신되는 데이터들과 설정치들의 차이를 쉽게 확인할 수 있다. 또한 격자에 수치들이 완전히 채워졌을 경우, '초기화 '버튼을 누르면 표시된 값들이 지워지게 된다. 마찬가지로 뉴메릭환경에서도 초기환경이나 그래프환경으로의 전환은 환경선택 메뉴나 툴바를 이용한다. 프로그램의 종료를 위해서는 초기화면의 파일메뉴에서 '종료'를 선택하거나 툴바의 가장 오른쪽에 위치한 것을 선택한다.

본 프로그램으로 압출성형기를 자동 제어하고자 할 경우, 시작과 끝은 반드시 초기환경에서 해야 하며, 그래프환경과 뉴메릭환경은 단지 수신되는 데이터를 사용자의 기호에 따라 표현할수 있도록 한 것이다.

3. 데이터 통신 방식(RS-485통신)

압출성형기를 자동 제어하는데 가장 중요한 것은 PC와 압출성형기간에 데이터들의 정확한 송·수신이라고 할 수 있다. 아무리 성능좋은 PC, 센서와 엑츄에이터를 사용한다 하더라도 데이터 송·수신 중에 그 값에 변화가 일어나거나 유실된다면 제어가 불가능하게 된다. 또한 데이터 송·수신거리가 짧아도 문제가 될 수 있다.

일반적으로 제어에 필요한 데이터를 송·수신하는데 이용되는 통신방식은 직렬(serial) 통신이다. 직렬 통신이라함은 직렬포트인 COM 포트를 통해 데이터를 송·수신하는 방식으로 병렬(parallel) 통신 방식에 비해 장거리 통신이 가능하고 경비가 싸다는 장점이 있어서 공장의 기계들을 자동화하는데 많이 이용되고 있다. 직렬 통신

을 하기 위해서는 어떠한 전기적인 규약이 필요한데 가장 널리 알려진 것으로는 RS-232C 가 있으며, PC내부에는 RS-232C 인터페이스가 장착되어 있기 때문에 PC의 직렬 통신도 이 규약을 따른다.

본 연구에서 데이터를 송·수신하는데 이용된 통신 규약은 RS-232C와 RS-485 이다. 즉 PC에서 명령을 내보내고 수신되는 값을 받아들이는 것은 RS-232C를 따르며, PC에서 나온 데이터 신호는 COM 포트에 연결된 변환 장치를 거치면서 RS-485방식으로 바뀌어 각각의 마이크로프로세서로 전달된다. 또한 마이크로프로세서에서 PC로 전달되는 데이터 신호는 역시 변환 장치를 거치면서 다시 RS-232C방식으로 변환되어 COM 포트로 수신된다.

RS-485방식은 산업현장에서 많이 쓰이는 방식으로 이상적인 경우 1.2 Km의 거리를 전송할 수 있을 정도로 외부의 노이즈에 강한 특징이 있다. 또한 데이터를 송·수신하는데 필요한 선이 2개 뿐이기 때문에 LAN 처럼 하나의 케이블을 통해 데이터를 주고 받는 버스형 네트워크를 갖출 수 있으므로 제어용 하드웨어의 배선을 단순화시킬 수 있다. 하지만, PC에는 직접 연결하여 사용할 수 없기 때문에 RS-232C 와 RS-485를 서로 상호 변환시킬 수 있는 어뎁터를 따로 구성해 주어야 한다.

4. 데이터 입력과 출력

사용자가 제어용 프로그램상에서 통신환경을 설정하고 포트를 열면 윈도우즈 95 환경에서는 통신 포트를 하나의 파일 개념으로 생각하여 CreateFile이라는 고유 함수를 사용하여 포트를 열게 된다. CreateFile 함수의 구성은 다음과 같다.

```
CreateFile(
        LPCTSTR      lpFileName,
```

```
        DWORD        dwDesiredAccess,
        DWORD        dwShareMode,
        LPSECURITY_ATTRIBUTES lpSecurityAttribute,
        DWORD        dwCreationDistribution,
        DWORD        dwFlagAndAttributes,
        HANDLE       hTemplateFile
    );
```

CreateFile 은 포트만 여는 데 사용하는 함수가 아닌 일반적인 파일이나 또는 디바이스를 여는 데도 사용하므로 여러 가지 인자가 있다. lpNameFile은 파일의 이름을 뜻하는데 여기서는 통신포트의 번호를 넣으면 된다. dwDesireAccess란 접근모드로서 COM포트에 명령을 쓰고 수신된 값들을 읽어야 하므로 GENERIC_READ | GENERIC_ WRITE 라고 한다. 보안에 관한 속성은 NULL로 해주며, dwCreationDistribution은 파일을 어떻게 열 것인가를 결정하는 인자로서 새로 만들 것인가 아니면 항상 기존의 파일을 열 것인가 등 여러 가지 인자가 있지만, COM포트는 존재하는 것이므로 새로 만들 필요 없이 기존의 것을 열어야 하므로 OPEN_EXISTING으로 하고, dwFlagAndAttribution은 파일 속성을 나타내는 인자로서 보통 파일이면서 overlapped되어야 하므로 FILE_ATTRIBUTE_NORMAL | FILE_FLAG_OVERLAPPED라고 설정한다. 마지막의 템플레이트 파일 핸들러는 필요없으므로 NULL로 설정한다. CreateFile 함수를 사용한 예는 아래와 같다.

```
CreateFile( “COM1",
            GENERIC_READ | GENERIC_WRITE,
            0,
            NULL,
            OPEN_EXISTING,
```

```
FILE_ATTRIBUTE_NORMAL |
FILE_FLAG_OVERLAPPED,
NULL );
```

위와 같이 포트를 열어 놓으면 PC는 송신상태로 대기하고 있으며, 서브컨트롤러는 수신상태로 대기하고 있다. 압출성형기를 가동하기 시작하면 타이머에 의해 일정한 시간 간격으로 공정변수와 시스템변수의 값을 보내라는 명령과 그에 따른 엑츄에이터를 가동하는 명령이 송신되고, 그 명령에 따라 각각의 명령은 8 바이트의 데이터 블록으로 되어 있다. 송신되는 데이터 블록은 아래와 같으며, 송ㆍ수신에 이용되는 address 및 command는 Table 3-7과 같다.

start byte
transmit address byte
receive address byte
command byte
high data byte
low data byte
check sum
stop byte

송신되는 데이터 블록에서 transmit address byte는 PC를 뜻하며 receive address byte는 각각의 마이크로프로세서를 뜻한다. command byte는 센서들의 값을 보내라는 명령 또는 엑츄에이터의 값을 변경시키라는 명령이다. high data byte 와 low data byte가 엑츄에이터를 변경시킬 실제 값이다.

Table 3-7. Commands for the communication between PC and subcontroller

Sensor & Acutator	ID	ID address	Command1[1]	Command2[2]	On/Off
Master or PC		0x10			
Motor 1	Main screw	0x01	0x03	0x02	
Motor 2	Feeder	0x02	0x03	0x02	
Motor 3	Cutter	0x03	0x03	0x02	
Thermocouple	Temp. 1	0x04	0x08		
	Temp. 2	0x04	0x81		
	Extrudate temp.	0x04	0x82		
Analog to digital	Die Pressure	0x05	0x90		
	Power	0x05	0x91		
Heater[3]	Heater 1	0x06		0x10	
	Heater 2	0x06		0x10	
Cooler[3]	Cooler 1	0x06		0x10	
	Cooler 2	0x06		0x10	
Water pump		0x07		0x10	

1) Command using in receving values of sensors through microprocessor
2) Command using in changing values of actuator through microprocessor
3) Certify suitable actuator by using 4 bits position among 8 bits of low_data byte

이러한 명령을 송신하려면 컴포트에서 output 시켜야 하는데, 여기에 사용된 함수가 'WriteCommBlock' 이며 사용예는 다음과 같다.

```
r_main[0]=0x55; r_main[1]=0x10; r_main[2]= 0x01;
r_main[3]=0x03; r_main[4]=0x01; r_main[5]= 0x01;
r_main[6]=0x10^0x01^0x03^0x01^0x01;
r_main[7]=0xaa;
m_pComm.WriteCommBlock(r_main , 8);
```

r_main[] 배열에 대응되는 값들은 r_main[0]에서부터 r_main[7]까지 차례대로 start byte, transmit address byte, receive address byte, command byte, high data byte, low data byte, checksum, stop byte에 해당하는 값들이고 위의 예는 메인스크류의 회전속도를 보내라는 명령을 보내는 예이다. WriteCommBlock의 인자 중에서 r_main은 보낼 데이터 블록의 시작 포인터이고, 8은 보내는 데이터의 총 바이트 수이다. WriteCommBlock은 통신포트를 통해 데이터를 송신하기 위한 사용자 정의 함수로서 실제로 데이터를 보내는 것은 WriteCommBlock 함수 내에 사용된 C 언어에서 제공되는 함수이며, 다음에 나타내었다.

```
BOOL CComm::WriteCommBlock( LPSTR lpByte , DWORD dwBytesToWrite)
{
        unsigned int      ASYNC_BASE;
        if(fConnected == TRUE){
                switch (bPort){
                case 4:           ASYNC_BASE=0x2e8;
                case 3:           ASYNC_BASE=0x3e8;
                case 2:           ASYNC_BASE=0x2f8;
```

```
            default:            ASYNC_BASE=0x3f8;
            }
            unsigned int  THR=ASYNC_BASE+0;
            unsigned int  MCR=ASYNC_BASE+4;
            unsigned int  LSR=ASYNC_BASE+5;

            for(DWORD i=0;i<dwBytesToWrite;i++)
            {
                _outp(MCR,0x0b); // /RTS->0
                _outp(THR,*(lpByte+i));
                do{/*...*/}while(!(_inp(LSR) & 0x40));
                _outp(MCR,0x09); // /RTS->1
            }
        }

        return ( TRUE ) ;

}
```

위의 내용 중에서 '_outp()'**는** output **포트에 한 바이트 단위의 데이터를 써주는 함수로서** conio.h **해더파일에에 정의되어 있으며 하드웨어를 직접 제어하는 함수이다.** **본래 윈도우즈** 95 **환경에서는 데이터를 파일에 쓰듯이** WriteFile**이라는 함수를 이용하여 포트에 데이터를 쓰는 것이 일반적이다.**

```
        WriteFile( idComDev, lpByte, dwBytesToWrite,
```

```
&dwBytesWritten, &osWrite ) ;
```

여기서, idComDev는 통신포트 번호를 뜻하고, lpByte는 써야할 데이터, dwByteToWrite는 써야할 데이터의 수, &dwBytesWritten은 써야할 바이트 수가 들어 있는 번지, &osWrite은 데이터를 쓰기 위한 구조체의 번지를 뜻한다. 하지만, 이러한 방법은 본 연구처럼 연속적으로 많은 양의 데이터를 주고 받을 경우 통신이 원할히 수행되지 않았기 때문에 포트를 직접 제어하는 _outp() 함수를 사용하였다.

이와 같은 과정을 통해 여러 가지 명령들이 마이크로프로세서로 전달되면 각각의 마이크로프로세서들은 수신 받은 데이터를 받아 해석하고 자신의 ID와 같으면 송신상태로 전환하여 수신 받은 데이터 블록의 내용에 따라 동작한다. 즉, 현재 센서의 값을 전달하라는 명령에 의해서는 그 값들을 PC로 보내고, 엑츄에이터의 값을 변경시키라는 명령에 대해서는 그 값을 변경시키면서 제어를 하게 된다. 이때 PC에 전달되는 데이터블록의 transmit address byte가 각각의 마이크로프로세서의 ID가 되며, receive address byte는 PC의 ID가 된다. PC의 포트에 단 하나의 바이트가 전달되더라도 포트에서 데이터를 받았다는 메시지가 전달되며 전달되는 데이터들을 버퍼 역할을 하는 char 형의 배열에 저장한 후, 각각의 데이터들을 get_char()라는 사용자 정의 함수에 의해 하나씩 데이터블록 구조에 맞는지 확인하는 과정을 거치면서 실제 센서의 값을 읽어들인다. 여기서 포트의 데이터를 읽는 역할을 담당하는 함수는 ReadFile이라는 함수로서 위에서 설명한 WriteFile의 짝이라고 말할 수 있다.

```
ReadFile( idComDev, lpszBlock,
          dwLength, &dwLength, &osRead ) ;
```

여기서, idComDev는 통신포트 번호를 뜻하고, lpszBlock 데이터

를 받는 버퍼의 주소, dwLength는 읽을 바이트의 개수, &dwLength는 읽을 바이트의 개수를 나타내는 숫자의 번지, &osRead는 데이터를 읽기 위한 구조체의 번지이다. ReadFile함수를 통해 포트에서 데이터를 읽어들이면 OnReceiveData()라는 사용자 정의 함수에 의해 중간버퍼에 저장이 된다.

```
LONG CMainFrame::OnReceiveData(UINT WParam,
                               LONG LParam)
{

        *(buff+start)=*(LPSTR)m_pComm.abIn;
        start++;
        start%=2000;
        return TRUE;

}
```

OnReceiveData()는 포트에 데이터가 들어올 때 마다 ReadFile()에 의해 전달되는 메시지에 의해 실행되며 메시지가 전달될 때 마다 데이터가 저장되는 배열 buff의 위치가 하나씩 이동하게 된다. 위에서 참고하면, 'start++'에 의해 buff의 위치가 변경된다. 이와 같이 버퍼에 데이터들이 저장되면 타이머에 의해 주기적으로 실행되는 사용자 정의 함수 GetData()에 의해 버퍼의 데이터를 하나씩 읽어들이면서 확인하고 화면에 나타낸다.

```
void CMainFrame::GetData()
```

```
{
    unsigned char t, data_h, data_l;
    while(start!=end){
    if (get_char()==0x55)
    {
    t=get_char();
    switch(t)
    {
    case 0x01:
        if(get_char()== 0x10)
        {
            t=get_char();
            switch(t)
            {
            case 0x03:
                data_h=get_char();
                data_l = get_char();
                com_main = 30*data_l;
                t = get_char();
                t = get_char();
                break;
            default:
                get_char();
                get_char();
                get_char();
                get_char();
```

```
                }
        }
                else{
                get_char();get_char();
                get_char();get_char();
                get_char();
        }
        break;
        ⋮
```

위에 나타낸 프로그램은 GetData()의 일부로서 내용을 살펴보면, 우선 get_char()라는 사용자 정의 함수에 의해 버퍼에 저장되어 있는 데이터를 하나씩 취하여 데이터블록 구조와 비교한다. 처음에 나온 'get_char() == 0x55'는 start byte인지 확인하는 단계이며, 다음에 get_char()하여 그 결과 값을 가지고 switch문을 통해 데이터를 보낸 마이크로프로세서가 어떤 것인지 선택한다. 'case 0x01:'은 메인스크류 구동 모터에 해당한다. 그 다음 'get_char() == 0x10'은 수신하는 것이 PC인지 확인하는 단계이며, 그 다음에 switch문을 통해 command를 확인하고 나서 2번 get_char()를 사용하여 data_h, data_l라는 변수에 그 값을 취하여 그 중에서 data_l에 해당하는 것을 메인스크류의 회전속도로 화면에 나타낸다. 메인스크류의 회전속도는 data_l만으로도 나타낼 수 있다. 그리고, checksum 과 stopbyte에 해당하는 값들은 단순히 get_char()함수를 한 번씩 사용함으로써 버퍼에서 읽어낸다. 만일 위의 내용 중에서 if 문이나 switch문에 의해 데이터블록을 확인하는 과정에서 일치하지 않는 경우가 발생하면 else 문이나 default문에 사용된 get_char()함수들에 의해 버퍼를 비워주게 된다. 아래는 get_char()함수에 대한 내용이다.

```
unsigned char get_char(void)
{
	unsigned char t;
	if(start==end){
		t=0x00;
	}
	else if(start!=end){
		t=buff[end];
		end++;
		end%=2000;
	}
	return t;
}
```

위의 과정을 거치면서 버퍼에서 읽어들인 데이터들은 화면상에 표시될 뿐만아니라 압출성형공정을 제어하기 위한 퍼지추론의 인자로서도 이용된다.

요약하여 말하면, 포트를 하나의 파일을 만들 듯이 CreateFile함수를 이용하여 열고, WriteCommBlock함수를 이용하여 포트에 명령데이터를 써주며, ReadCommBlock함수에 의해 포트에서 데이터를 읽어들이고, OnReceveData함수를 이용하여 버퍼용 배열에 저장한 다음, GetData함수에 의해 하나씩 확인하면서 실제 데이터만을 화면에 나타내고 제어에 사용한다. 이러한 과정이 반복되면서 압출성형공정을 자동제어하게 된다.

5. 인공지능제어 알고리즘

가. 자동 start-up 알고리즘

자동제어에 의한 start-up은 Fig. 3-21에서 보인 것과 같이 압출물의 온도(다이부근)가 일정한 수준까지 도달할 때까지 배럴의 온도를 상승시킨 후 냉각수를 작동하면서 냉각을 하고 다시 온도가 낮아지면 히터를 가동시켜서 높이는 방법을 반복하여 정상상태에 도달하도록 프로그램을 설계하였다. 여기서 일정한 수준이란 단순제어의 반복된 실험을 통하여 얻은 결과를 토대로 하였다.

먼저 압출물의 온도가 다이쪽의 최종배럴의 목표치에 도달할 때까지 히터를 on하고 그 이상이 될 때에는 히터를 off하면서 냉각수 밸브를 on하였다. 이와 같은 과정이 반복이 되면서 압출물의 온도 변화폭이 작아지면 단순히 배럴의 온도를 목표치에 수렴하도록 히터와 냉각수 밸브를 on/off제어 하였다. 최종적으로 배럴의 온도 변화폭이 작아지면 정상가동 단계에 도달한 것으로 간주하였다. 이때 구동모터회전속도, 원료공급속도, 물공급속도는 모두 30초 동안 선형적으로 목표치에 도달하도록 제어하였다. 다이의 압출물 온도가 일정한 값에 도달하기 오래 전에 다른 시스템변수인 압력과 SME는 일정한 값에 수렴한다. 단 전력이 극도로 높은 경우가 발생하면 언제라도 비상사태로 간주하여 자동 shut-down의 단계로 넘어가도록 하였다. 이에 관한 프로그램의 흐름도는 Fig. 3-22와 같다.

나. 자동 정상가동 알고리즘

압출성형기가 정상가동에 들어간 직후에 측정된 시스템변수를 일정하게 유지하기 위하여 공정변수를 퍼지추론(fuzzy reasoning)방법을 적용하여 제어하였다(13-19). 각 시스템변수는 퍼지값으로, 공정변수는 확정값(deterministic value)로 설정하여 정상가동중 시스템변수가

입력되면 그에 대응하는 공정변수의 제어값들이 퍼지추론을 통하여 출력되어 압출성형기를 제어하는 과정이 반복되도록 하였다. 퍼지추론의 세부과정은 다음과 같다.

(1) 퍼지값(fuzzy values)의 설정

퍼지법칙을 작성하기 위한 전처리 작업으로 관련 변수들에 대한 퍼지 값을 설정하였다. 퍼지값에서는 소속함수에 의해 어떤 성질에 대한 소속된 정도를 나타낸다. 본 연구에서는 스크류 회전속도, 배럴 온도, 물투입량등의 공정변수의 변화량과 이에 따라 변하는 압출물의 시스템변수를 퍼지 변수로 선택하였으며, 이들 변수에 대해 임의의 성질을 HIGH, SAME, LOW의 3단계로 구분지었으며, 이에 대한 퍼지값들은 다음 식과 Fig. 3-23에 나타낸 것과 같은 사다리꼴 또는 삼각형의 소속함수로 결정하였다.

$$u = \begin{cases} 0 & (x < a1) \\ (x - a1)/(a2 - a1) & (a1 \le x < a2) \\ 1 & (a2 \le x < a3) \\ (a4 - x)/(a4 - a3) & (a3 \le x < a4) \\ 0 & (a4 \le x) \end{cases}$$

$$P(X) = \{ (x, u) \mid x \in X, u \in [0, 1] \}$$

여기서, a1, a2, a3, a4는 퍼지값내 집합값의 범위를 나타내는 경계값이며, P(X)는 퍼지값을 나타낸다.

실제 설정된 시스템변수의 퍼지값은 Tables 3-4, 3-5와 Figs. 3-3 3-11의 결과로부터 경험적 지식에 근거하여 임의로 설정하였다(Fig. 3-24). 각각의 멤버십 함수들은 원만한 제어를 위해 일정 수준의 오차를 인정하는 사다리꼴의 형태를 취하였으며, 퍼지 값은 예비

실험을 거치면서 얻은 경험적 수치를 바탕으로 설정하였으며 공정변수의 조건에 따라 나타나는 시스템변수의 변화량을 기준으로 하였다.

(2) 퍼지법칙

퍼지법칙은 얻어진 전문가 지식을 근거로 하여 관련된 변수에 대한 퍼지 값들 사이의 관계를 나타내는 것으로 다음과 같이 IF THEN …의 문장형식으로 나타내었다.

IF P(Y1) AND P(Y2) AND P(Y3) THEN P(X1) AND P(X2) AND H/C
IF P(Y1) AND P(Y2) AND P(Y3) THEN P(X1) AND P(X2) AND H/C
IF P(Y1) AND P(Y2) AND P(Y3) THEN P(X1) AND P(X2) AND H/C
⋮ ⋮

여기서, P(Y1), P(Y2), P(Y3)는 각각 압출성형공정의 공정변수에 따라 나타나는 시스템변수들이며, P(X1), P(X2), H/C는 압출성형공정을 제어하기 위한 조절변수들이다. 본 연구에서는 예비실험을 통해 구축한 전문가 지식과 퍼지값을 이용하여 퍼지 법칙을 설정하였다.

Fig. 3-24에서와 같이 3가지 시스템변수를 동시에 제어하기 위해 작성한 전문가 지식은 Table 3-8과 같다. 이것은 시스템변수에 대한 임의의 성질을 HIGH, SAME, LOW 의 3 단계로 구분하여 전체 27가지의 경우 중에서 발생할 수 있는 경우인 21가지만을 체택하였다. 즉, ΔSME, Δ(die pressure), Δ(extrudate temp.) 가 모두 H인 경우(Table 3-8 의 첫 번째 경우), SME에 가장 영향을 크게 미치는 물의 투입량을 크게 증가시키고, 압출물의 온도가 높은 상태이므로 히

터를 끄고, 냉각수를 흘려보낸다. 이때 물 투입량 증가와 히터, 냉각수 변화에 의해 갑작스럽게 SME의 변화를 방지하고자 스크류의 회전속도는 약간 증가시킨다. 두 번째 경우에는 높은 상태인 SME와 압력을 낮추기 위해, 물투입량을 증가시키고 이에 대해 압출물의 온도가 낮아지는 것을 방지하고자 히터를 켜고, 스크류의 회전속도도 증가시킨다. 세 번째 경우에는 SME 와 압력이 높은 상태이지만, 압출물의 온도가 낮기 때문에 물투입량을 증가시킬 수는 없고 대신 단지 히터를 켜줌으로써 압출물의 온도를 상승시킨다. 네 번째 경우에는 압력은 일정한 상태이지만, SME와 압출물의 온도가 높은 상태이므로, 물 투입량을 증가시키고 히터를 끈다. 이때 압력이 떨어지는 것을 방지하고자 스크류의 회전속도는 감소시킨다. 다섯 번째 경우에는 SME만 높은 상태이므로 물 투입량을 약간 증가시키고 이때 압력의 하강과 온도가 떨어지는 것을 방지하고자 히터는 켜고 스크류의 회전속도는 감소시킨다. 여섯 번째의 경우에는 압력이 낮은 상태이므로 물 투입량은 증가시킬수 없고, 히터를 끄고, 스크류의 회전속도를 감소시킴으로써 높은 상태인 SME와 압출물의 온도를 떨어뜨린다. 일곱 번째의 경우 압력과 온도가 높은 상태이므로 물 투입량을 약간 증가시키고, 히터는 끈다. 이때 SME가 떨어지는 것을 방지하고자 스크류의 회전속도는 증가시킨다. 여덟 번째의 경우 압력만 높은 상태이므로 물투입량을 약간 증가시켜 떨어뜨리며 그에 따라 SME와 온도가 떨어지는 것을 방지하고자 스크류의 회전속도를 증가시킨다. 아홉 번째의 경우 낮은 온도를 상승시키기 위해 히터를 켜고 스크류의 속도를 증가시킨다. 열 번째의 경우 온도를 떨어뜨리기 위해 물 투입량만 약간 증가시킨다. 열 한 번째의 경우에는 모두 일정한 수준이므로 아무런 값을 변화시키지 않는다. 나머지 것들에 대한 것은 열한 번째를 기준으로 하여 지금까지 설명한 것의 반대 경우에 해당한다. 이에 대한 조절변수인 물 투입량, 스크류의 속도, 히

터 및 냉각수는 각각 4수준, 1수준, 1수준으로 구분지었다.

(4) 입력값과 퍼지법칙의 합성

설정된 퍼지법칙은 연산법에 의하여 다음 식의 relation matrix, R로 변환되는데 R의 각 성분에 대한 소속함수는 $u_R(y1, y2, y3, xi,)$ 이다.

$$R = P(Y1) \times P(Y2) \times P(Y3) \times P(Xi)$$

$$u_R(y1, y2, y3, xi) = \min [u_{y1}(y1), u_{y2}(y2), u_{y3}(y3), u_{xi}(xi)]$$

여기서, y1, y2, y3는 각각 SME, 다이 압력, 압출물의 온도이며, xi는 메인스크류의 속도 및 물 투입량의 조절을 나타낸다.

측정변수로서 압출성형 공정의 시스템변수가 측정되면, 제어를 위한 메인스크류의 회전속도, 물투입량의 변화량이 다음과 같이 산출된다. 퍼지 법칙에서 산출된 relation matrix R과 임의의 입력 퍼지값, P(Y1'), P(Y2'), P(Y3')를 Mamdani방식으로 합성하여 그에 대응하는 출력값 퍼지값, P(Xi')을 다음 식과 같이 얻는다.

$$P(Xi') = [P(Y1') \times P(Y2') \times P(Y3')] \circ R$$

$$u_{xi}(xi') = \max [\min (u(y1', y2', y3'), u_R (y1, y2, y3, xi))]$$

여기서, 임의의 입력 퍼지 값들은 측정되는 확정적 값인 y1', y2', y3'을 퍼지 법칙에 사용된 퍼지 값의 소속함수에 대입하여 산출하였다.

(5) 출력값의 비퍼지화

위와 같은 과정을 통해 산출된 출력값은 그 형태가 퍼지값으로 되어있기 때문에 압출성형기를 제어하기 위해서 확정적인 값으로 변

환되어야 하는데, 그 방법으로는 center of gravity of fuzzy set 또는 mean of all the maxima of fuzzy set등의 방식이 있는데, center of gravity of fuzzy set 방식에 의해 다음 식에 의해 산출하였다.

$$C = \sum_{i=1}^{n} xi\ ux1(y1i) \ / \ \sum_{i=1}^{n} ux1(y1i)$$

여기서 C는 출력된 퍼지 값 { y1i ∈ Y1 | ux1(y1i) ∈ [0, 1] }을 비퍼지화하여 얻은 하나의 수치값이다.

지금까지의 퍼지 추론 과정을 요약하면, Fig. 3-25과 같다.

(6) 퍼지추론 프로그램

Fig. 3-24과 Table 3-8의 결과를 바탕으로 압출성형 정상가동을 제어하는데 필요한 조절변수를 산출하는 퍼지추론 프로그램을 작성하였고 그 흐름도는 Fig. 3-26과 같다. 즉, 정상가동 중에 SME, 다이부근의 압력, 압출물의 온도등이 측정되어 퍼지추론 프로그램으로 입력되면, 메인스크류의 회전속도, 물 투입량, 히터 및 냉각수의 on/off 를 조절하는 조절변수가 산출되어 압출성형 정상가동을 제어하는 과정이 반복된다.

본 퍼지추론 프로그램은 압출성형 정상가동 제어용 프로그램에 삽입되어 서브 루틴 함수로서 이용되었다.

다. 자동 shut-down 알고리즘

자동제어에 의한 shut-down은 Fig. 3-27에서 보인 것과 같이 shut-down시 전력의 증감치를 측정해 보아 feeder의 투입량을 결정

하는 방식으로 정하였다. 즉, 배럴의 히터는 shut-down 시작과 함께 off하고 물공급속도는 20초 내에 초기목표치의 약 6배 까지 선형적으로 증가시켰으며 구동모터회전속도는 20초 후에 3초내에 초기목표치의 1/6로 감소시킨 후 일정하게 유지하였다. 배럴내 충전물의 수분/고형물의 비율을 조절하기 위해 위와 같이 물공급 속도는 단시간내에 최대로 일정하게 해놓고 원료공급속도를 주요 변수로 간주하여 퍼지 추론방식을 적용하여 제어하였다. 전문가 지식으로서의 막힘현상은 shut-down이 진행되면서 측정되는 구동모터의 전력이 급격하게 감소할 경우 발생가능성이 매우 높으므로 원료공급속도의 감소를 멈추고, 전력이 어느정도 일정할 경우와 완만하게 감소할 경우에는 배럴내의 충전물이 스크류에 의해 사출구 밖으로 밀어내는 힘을 받는데 양호한 상태에 있게 되므로 원료공급속도를 감소시키는 방법을 반복하여 shut-down이 진행되도록 프로그램을 설계하였다.

먼저 shut-down이 시작함과 동시에 원료공급속도가 감소되면 그에 따른 전력이 감소함을 볼 수가 있는데 전력감소의 변화량이 어느 수준 이상이 되거나 전력이 감소하지 않고 오히려 증가하려는 추세가 보일떠에는 원료공급속도를 일정하게 유지시킨다. 이러한 상태에서 약간의 시간이 경과하면 전력은 다시 완만히 감소하게 되며 이 때에 원료공급속도를 감소시키게 되는데 이와 같은 과정이 반복이 되면서 원료공급모터는 결국 멈추게 하였다. 상기의 전문가 지식에 관한 퍼지법칙을 Table 3-9과 같이 작성하였고 시간간격을 달리한 전력의 변화량은 퍼지값으로, 이에 따른 원료공급속도는 확정값으로 설정하였다. 즉 Shut-down중 전력이 입력되면 퍼지추론을 통하여 그에 대응되는 원료공급속도가 출력되어 압출성형기를 제어하는 과정이 반복되며 이 때 퍼지값의 설정은 경험에 근거하여 임의로 설정하였다 (Fig. 3-28). 이에 관한 프로그램의 흐름도는 Fig. 3-22와 같다.

제4절 Case studies

1. 자동 shut-down

인공지능제어 stat-up을 한 결과는 Fig. 3-29과 Fig. 3-30에 나타내었다. 즉, 물공급속도 9rpm, 원료공급속도 54rpm, 스크류구동모터 180rpm, 배럴 1, 2의 온도 120, 100℃의 목표치를 갖고 자동 start-up했을떄 SME와 압력은 약 1분 경에, 다이부근의 압출물의 온도는 약 7분 경부터 일정한 값으로 수렴하여 정상가동에 도달하였다. 즉 다이부근의 압출물의 온도가 설정치에 다다랐을 때 배럴 1의 온도는 200℃를 훨씬 넘는 높은 온도에까지 올랐으며 이 때에 SME는 한 때 높게 측정되기도 하였으나 overshoot될 정도는 아니었으며 대체적으로 안정한 경향을 유지시키며 진행하였다 (Fig. 27). 또한 물공급속도 8rpm, 원료공급속도 59rpm, 스크류구동모터 180rpm, 배럴 1, 2의 온도 120, 100℃의 목표치를 갖고 자동 start-up했을떄 SME와 압력은 case study Ⅰ의 경우와 마찬가지로 약 1분 경에, 다이부근의 압출물의 온도는 약 7분 경부터 일정한 값으로 수렴하여 정상가동에 도달하였다. 즉 다이부근의 압출물의 온도가 목표칭 다다랐을 때 배럴 1의 온도는 200℃를 넘지만 case study Ⅰ의 경우보다는 약간 낮은 온도에 까지 올랐으며 이 때에 SME는 크게 높이 오르는 일 없이 case study Ⅰ의 경우보다도 더 안정하게 유지하며 진행하였다 (Fig. 28).

이와 같이 start-up이 자동으로 진행되어 구동모터 power의 overshoot가 일어나지 않고 시스템변수가 단시간내에 도달함을 알 수 있었다.

2. 자동 정상 가동

가. 원료에 투입되는 물의 양이 불균일할 경우

메인스크류의 회전속도 180 rpm, 원료 공급속도 1500 rpm, 배럴의 온도가 100 120℃, 물 공급속도 8 rpm의 공정조건에서 정상가동되는 중, 비정상상태를 의도적으로 만들어주기 위해, 원료 투입 부위에 물을 일정시간 공급하여 그 제어되는 양상을 관찰하였다. 이때 추가적으로 공급된 물의 양은 20 ㎖/min의 속도로 약 5 분간 공급하였을 때, 시스템변수인 SME, 다이부근의 압력, 압출물의 온도가 모두 감소하였다. 이때 이러한 현상에 대한 제어동작으로 공정변수인 물 투입량이 감소하고 메인스크류의 회전속도와 배럴온도가 증가되어 시스템변수 값이 단시간 내에 원래 상태로 회복되었다. 또한 물의 추가 공급을 중지하였을 때, 다시 공정변수가 원래의 상태를 회복하면서 시스템변수 역시 최초의 상태로 회복되었다. Fig. 3-31는 위의 과정 중에 수집된 데이터를 그래프로 나타낸 것이다.

나. 원료의 공급속도가 불균일할 경우

메인스크류의 회전속도 180 rpm, 원료 공급속도 1500 rpm, 배럴의 온도가 100 120℃, 물 공급속도 8 rpm의 공정조건에서 정상가동되는 중, 원료의 투입량이 갑자기 줄어드는 비정상상태를 의도적으로 만들어주기 위해, 원료 공급속도를 1000 rpm으로 일정시간 늦춘 후 그 제어되는 양상을 관찰하였다. 이때 원료 공급속도 제한으로 원료의 양은 정상가동시 보다 약 138 g/min 줄어든 양이 공급되었으며, 그 기간은 약 5분간이었다. 투입되는 원료가 줄어들면서, 내부의 충진정도가 감소함에 따라 다이부근의 압력이 급속히 떨어졌으며, SME와 압출물의 온도는 상승하는 현상을 보였다. 이러한 현상에 대한 제어동작으로 메인스크류의 회전속도가 감소하였으며 물 투입량은 약간 증가하고, 배럴의 온도는 떨어지면서 시스템변수가 원래의 상태로 회복되었다. 또한 원료의 공급속도를 다시 설정치에 맞게 공급하였더니 공정변수가 원래의 상태로 회복되면서 시스템변수도 최

초의 상태로 되돌아 갔다. Fig. 3-32는 위의 과정 중에 수집된 데이터를 그래프로 나타낸 것이다. 이와 같이 정상가동 중, 원료조건등의 불균일한 경우가 발생하더라도 압출물의 물성의 지표가 될 수 있는 시스템변수가 일정하게 제어됨을 알 수 있었다.

3. 자동 shut-down

인공지능제어 shut-down을 한 결과는 Fig. 3-33과 Fig. 3-34에 나타내었다. 즉, 물공급속도 8rpm, 원료공급속도 50rpm, 스크류구동모터 180rpm, 배럴 1, 2의 온도 120, 100℃의 정상가동 중에 자동 shut-down을 실행시키면 52rpm의 목표치를 향해 선형적으로 물공급속도를 증가시키고 이에 맞추어 구동모터의 회전속도를 150rpm으로 감소시켜 측정되는 구동모터 power의 변화량에 따라 원료공급속도가 계단식으로 제어되면서 막힘의 현상이 없이 안전하게 가동이 종료되었다 (Fig. 33).

또한 물공급속도 58rpm의 목표치를 설정하고 다른 공정변수들은 case study Ⅰ 과 마찬가지로 하여 shut-down을 실행시켜본 결과 case study Ⅰ 과 마찬가지로 막힘의 현상이 없이 안전하게 종료됨을 알 수 있었다 (Fig. 34). 다만 case study Ⅰ의 경우에는 약 3단계의 계단식으로 제어된 반면 case study Ⅱ 의 경우에는 2단계의 계단식으로 제어되었음을 알 수 있는데 이는 물공급속도의 차이때문일 것으로 생각된다.

이와 같이 모든 경우에서 shut-down이 자동으로 진행되어 막힘 현상이 일어나지 않고 단시간내에 가동이 종료됨을 알 수 있었다.

제 4 장 결 론

먼저 본 연구의 최종 결과를 정리해 보면 다음과 같다. 1) 인공지능제어에 의하여 압출성형기의 start-up 과정에서 원료투입량, 수분첨가량, 히터 및 냉각기, 스크류 회전속도가 자동으로 제어되어 빠른 시간 내에 정상가동 상태에 도달한다. 2) 정상가동 중 원료의 투입량 및 배합비 등의 변화(fluctuation)에 의한 압출물의 품질변화를 시스템변수의 제어에 의해 일정하게 유지할 수 있으며 비상 사태에 자동으로 shut-down시킬 수 있다. 3) Shut-down 과정에서 압출성형기의 막힘현상이 일어나지 않도록 제어하여 작동을 빠른 시간내에 종료할 수 있다. 결론적으로 압출성형공정 전문가가 아닌 초보자라 할지라도 그 사용이 쉬운 컴퓨터의 프로그램을 조작함으로서 압출성형기를 사용할 수 있게 되었다.

본 연구의 근본적인 취지는 현장애로기술에 활용하기 위함이었다. 국내의 열악한 현황에서 본 기술을 적용할 경우 파생될 수 있는 긍정적인 효과에 대하여 다음과 같이 생각해 볼 수 있다. 1) 자동제어 압출성형기의 수입 대체 및 수출 가능성을 기대할 수 있다. 특히 원화 가치의 하락으로 인하여 국산화의 좋은 기회가 될 것이다. 2) 사료 생산 초보자도 압출성형기를 손쉽게 작동할 수 있게 되고 사료제품의 품질이 향상되므로 기계 판매 시장성이 높아진다. 3) Start-up, shut-down 시간의 단축으로 원료 손실량을 절감할 수 있다. 4) 자동제어로 인한 노동력을 감축하고, 에너지 효율 및 생산성을 향상시킬 수 있다. 5) 사료 생산업체의 기술의 낙후성에도 불구하고 인공지능제어 압출성형기는 스스로 효율적으로 작동할 수 있으므로 비용 절감 및 사료의 품질이 향상되어, 파생적인 효과로서 농·어가에서 동물을 사육할 때 사료의 원가 절감 및 고품질에 따른 전반적인 능률의 향상을 기대할 수 있다.

또한 상기 기술의 실용화를 위하여 국내 압출성형기 제작업체에 직접 적용하여 국내·외 압출성형기의 기계 판매 시장성을 높이는데 활용해야 할 것이다. 이에 대한 세부계획은 다음과 같다. 1) 압출성형기의 본체를 기존 기술로 제작한다. 2) 제어에 사용되는 하드웨어를 본 연구결과를 적용하여 제작한다. 3) 제어에 사용되는 소프트웨어 프로그램을 역시 적용하고 이를 바탕으로 사료뿐만 아니라 식품을 대상으로도 소프트웨어를 개발한다. 4) 기계 소비자의 주문에 맞는 사양을 고려하여 제어 방식을 보다 다양하게 개발한다. 5) 본 기술을 기타 농·축·수산 가공 기계의 자동화에도 응용한다. 또한 해당 기술에 의해 제작된 압출성형기를 국내 사료 생산업체에 보급할 때 경제적인 부담을 절감할 수 있도록 정부 보조금을 지원 받는다면 고성능의 기계를 값싸게 공급하여 사료 생산업계의 활성화를 기대할 수 있다.

참고문헌

1. Frame N. : Pet and Aquatic Feed Production, Summitted as a Chapter in Textbook Extrusion cooking. Wenger Manufacturing (1993)

2. Harper J. M. : Food Extruders and Their Application, In Extrusion Cooking. p.1-16 American Association of Cereal Chemists (1989)

3. Van Zuilichem D. J. and Stolp W. : Survey of the Present Extrusion Cooking Techniques in Food and Confectionary Industry, In Extrusion Technology for The Food Industry. p.1-15. Elsevier Applied Science. (1986)

4. Eerikäinen T., LinkoS. and Linko P. : The Potential of Fuzzy Logic in Optimization and Control : Fuzzy Reasoning in Extrusion Cooker Control, In Automatic Control and Optimization of food processes. Paris, France. p.183-201. (1986)

5. Wiedmann W. and Strecker J. : Process Control of Cooker-Eextruder, In Automatic Control and Optimization of Food Processes. p.201-214. Elsevier Applied Science. (1986)

6. Meuser F. and Wiedmann W. : Extrusion Plant Design, In Extrusion Cooking. p.91-156. American Association of Cereal Chemists (1989)

7. Bouvier, J. M., Gelus, M., Martin, A., and Sanei, A. Reproducibility in Extrusion-Cooking Texturization. Application to Extruded Biscuits(Abstr.) Kemia-Kemi 14(10b) 1052

8. **한 억**, **이상효**, **이현유**, **김영명**, **민병용** : **압출성형에 의한 알파 미분의 물리화학적 특성**, **한국식품과학회지**, 20(4) 470-475 (1988)

9. **송재철**, **박현정** : **식품물성학**, p.221-222. **울산대학교 출판부** (1995)

10. Park J., Rhee K. S., Kim B. K. and Rhee K. C., Single-Screw Extrusion of Defatted Soy Flour, Corn Starch and Raw Beef Blends, J. Food Sci. 58(1), 9 (1993)

11. Hilton J. W. , Cho C.Y. and Slinger S. J. : Effect of Extruder Processing and Steam Pelleting Diets on Pellet Durability, Pellet Water Absorption, and the Physiological Response of Rainbow Trout. Aquaculture. 25, 185 (1980)

12. Wiedmann W. and Strecker J. : How to Automate an Extruder, p.40-46. Food Eng. Int (1987)

13. Zimmermann, H.J. : Fuzzy Set Theory and its Applications. Kluwer Academic Publishers, Massachusetts (1991)

14. Klir, G.J. and Folger, T.A.: Fuzzy Sets, Uncertainty, and Information. Prentice-Hall International, Inc., London (1988)

15. Zhang, Q. and Litchfield, J.B. : Applying fuzzy mathematics to product development and comparison. Food Technol., 45, 108 (1991)

16. Singh, R.K. and Ou-Yang, F. : Knowledge-based fuzzy control of aseptic processing, Food Technol., 48, 155 (1994)

17. Eerikainen, T., Linko, S. and Linko, P. : The potential of fuzzy logic in optimization and control: fuzzy reasoning in extrusion cooker control. In Proceedings of the International Symposium on the Automatic Control and Optimisation of Food Processes, Paris, France, 12-13 November, p.183 (1986)

18. 변증남 : 퍼지(Fuzzy) 이론의 응용. 식품 Extrusion 기술 하계강좌, 한국식품 Extrusion 연구회, p.66 (1991)

19. 이승주, 원치선, 한억, 목철균, 이병상 : 압출성형공정 퍼지제어기의 모의실험. 한국식품과학회지, 27(2), 164 (1995)

부 록

컴퓨터 프로그램 코드

주제어기 Visual C++ **프로그램 코드**

부제어기 모터제어 C **와 어셈블러 프로그램 코드**

온도측정 C **와 어셈블러 프로그램 코드**

아날로그(압력, 파워, 유량)측정 프로그램 코드

히터, 냉각기, 물 공급 펌프 제어 프로그램 코드

주 의

1. 이 보고서는 농림부에서 시행한 농림수산특정 연구사업의 연구보고서입니다.

2. 이 보고서 내용을 발표할 때에는 반드시 농림부에서 시행한 농림수산 특정연구사업의 연구결과임을 밝혀야 합니다.

3. 국가과학기술 기밀유지에 필요한 내용은 대외적으로 발표 또는 공개하여서는 아니됩니다.

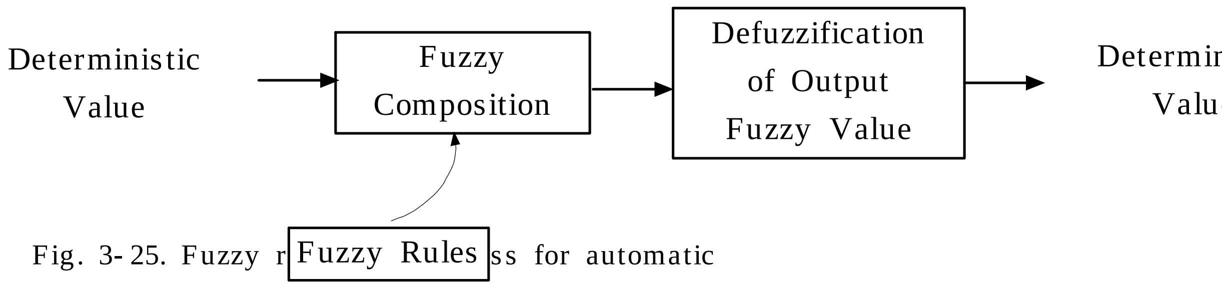


Fig. 3-25. Fuzzy r … ss for automatic control of extrusion process

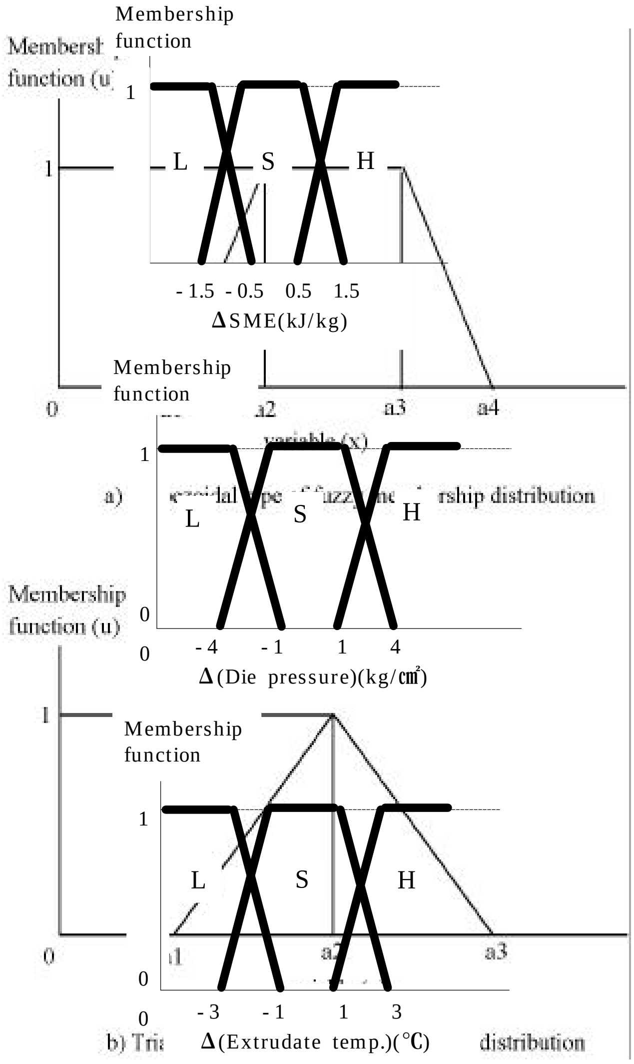


Fig. 3-23. Typical types of fuzzy membership distribution

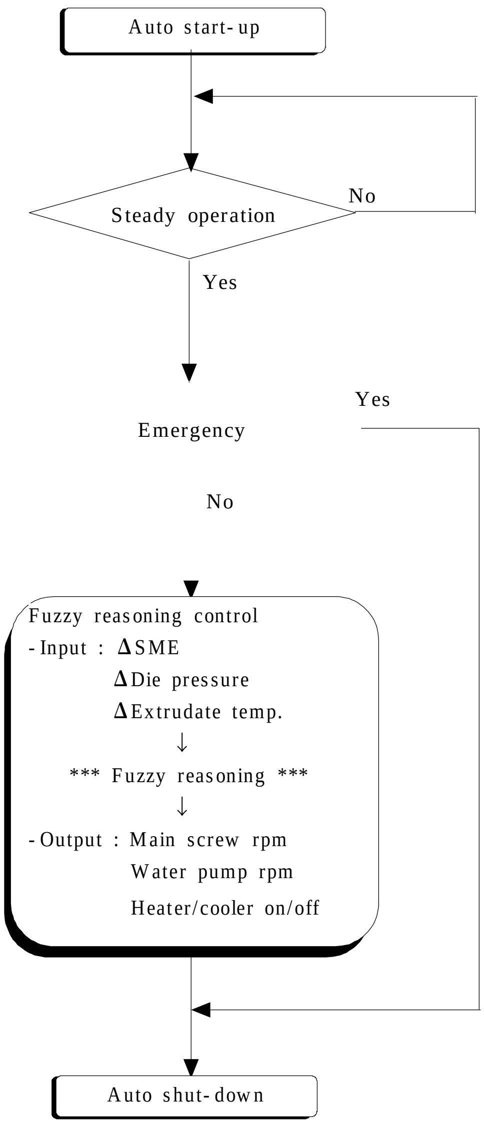
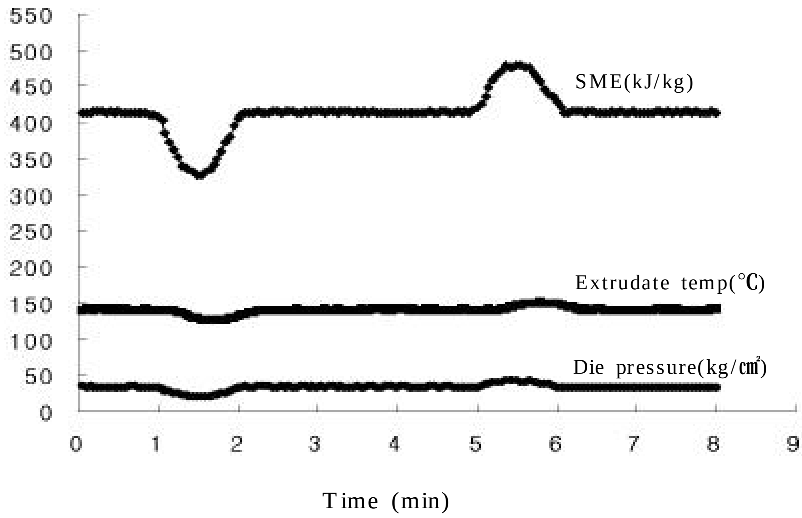
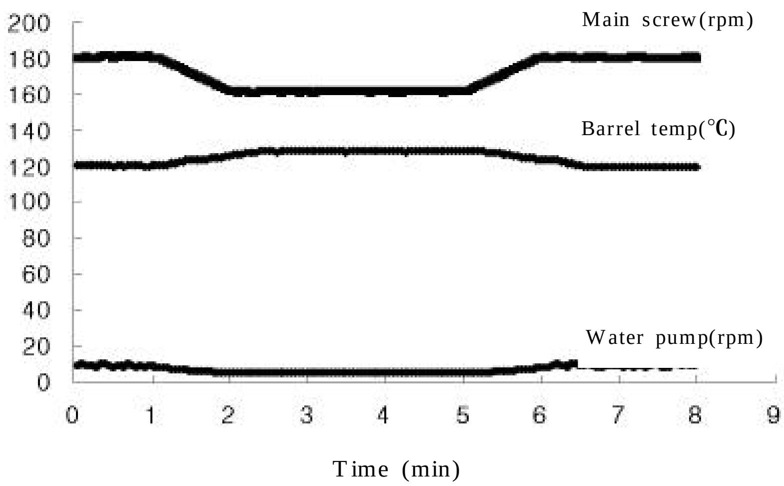


Fig. 3-24. Fuzzy values of ΔSME, Δ(die pressure), and Δ(extrudate temperature)

Fig. 3-26. Flow chart of computer program for automatic control of steady-operation

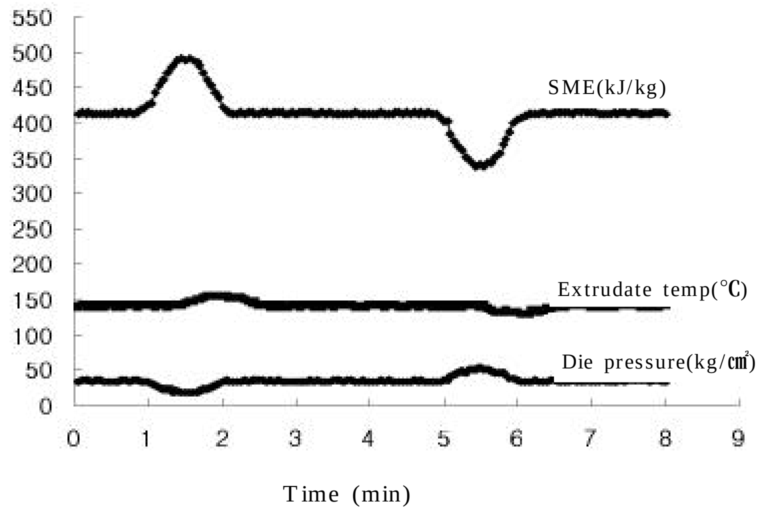


a) System variables in automatic control of extruder

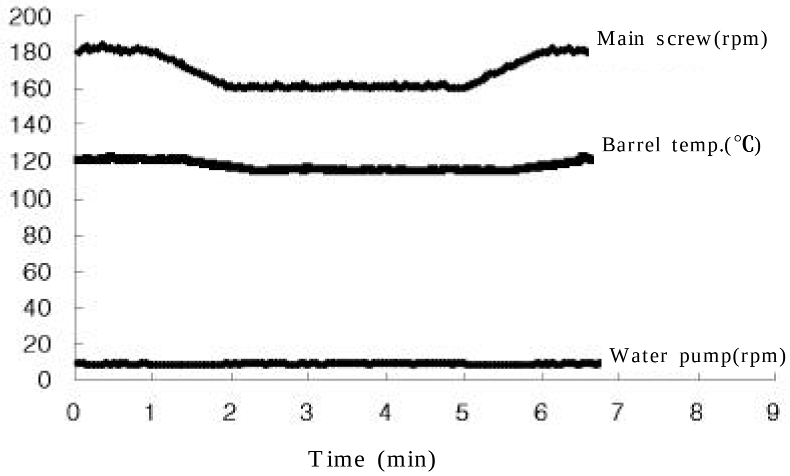


b) Process variables in automatic control of extruder

Fig. 3-31. Control case of automatic steady-operation of extruder with disturbance of excess water content



a) System variables in automatic control of extruder



b) Process variables in automatic control of extruder

Fig. 3-32. Control case of automatic steady-operation of extruder with disturbance of insufficient throughput

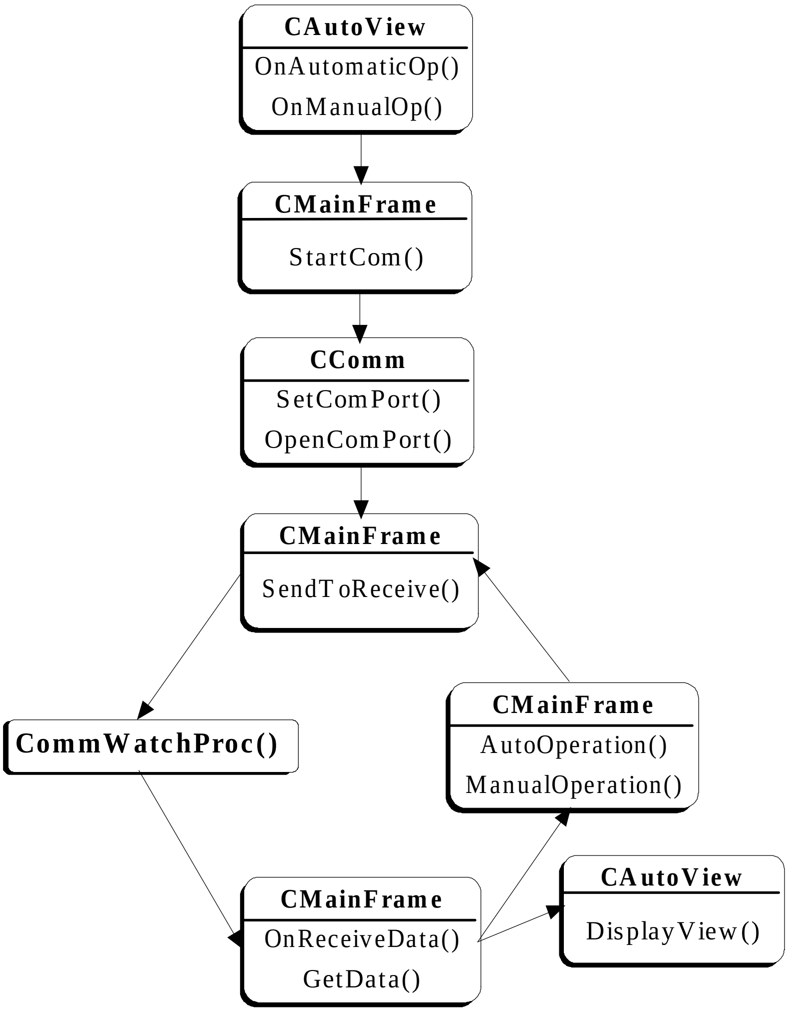


Fig. 3-14. Main loop of extruder automation program

Table 3-8. Fuzzy expert knowledge for steady-operation of extruder

IF	ΔSME	ΔDie pressure	ΔExtrudate temp.	THEN	Heater / cooler	Main screw	Water input
1	H[1]	H	H		↓[4]	↑[5]	↑↑↑↑
2	H	H	S[2]		↑	↑	↑↑
3	H	H	L[3]		↑	-	-
4	H	S	H		↓	↓	↑↑↑
5	H	S	S		↑	↓	↑
6	H	L	H		↓	↓	-
7	S	H	H		↓	↑	↑↑
8	S	H	S		-	↑	↑
9	S	H	L		↑	↑	-
10	S	S	H		-	-	↑
11	S	S	S		-	-	-
12	S	S	L		-	-	↓
13	S	L	H		↓	↓	-
14	S	L	S		-	↓	↓
15	S	L	L		↑	↓	↓↓
16	L	H	L		↑	↑	-
17	L	S	S		↓	↑	↓
18	L	S	L		↑	↑	↓↓↓
19	L	L	H		↓	-	-
20	L	L	S		↓	↓	↓↓
21	L	L	L		↑	↓	↓↓↓↓

1) H : HIGH
2) S : SAME
3) L : LOW
4) ↑ : ON or LEVEL UP
5) ↓ : OFF or LEVEL DOWN

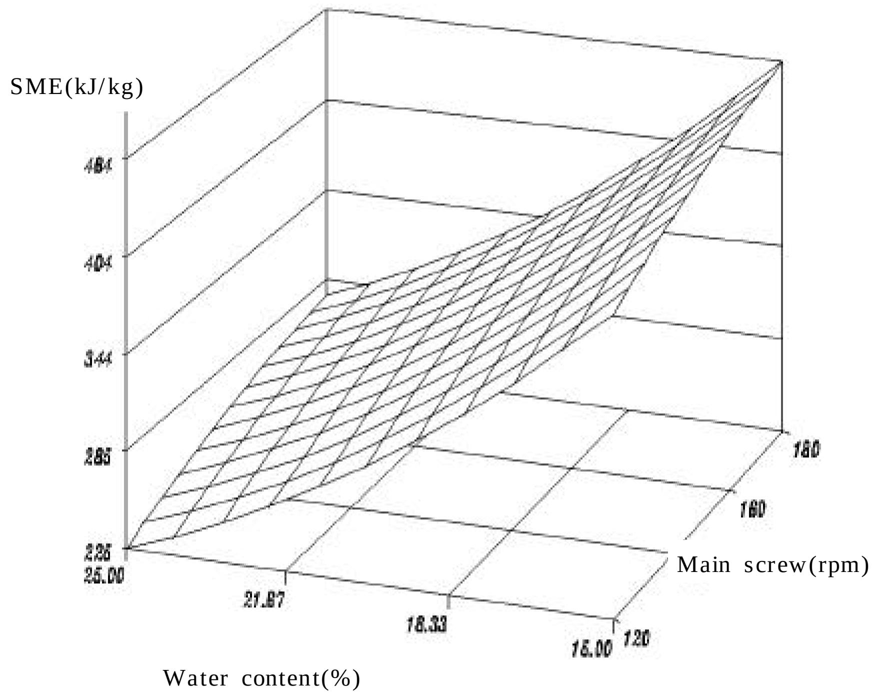


Fig. 3-3. Response surface of SME in terms of water content and screw speed at 1, 2 barrel temperature 60, 80℃

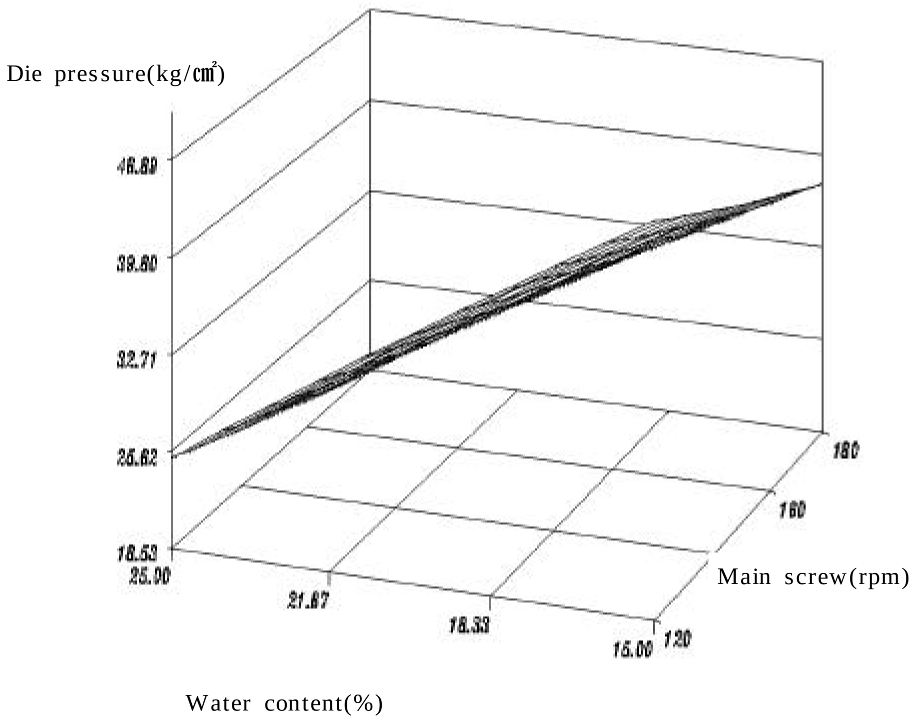


Fig. 3-4. Response surface of die pressure in terms of water content and screw speed at 1, 2 barrel temperature 60, 80℃

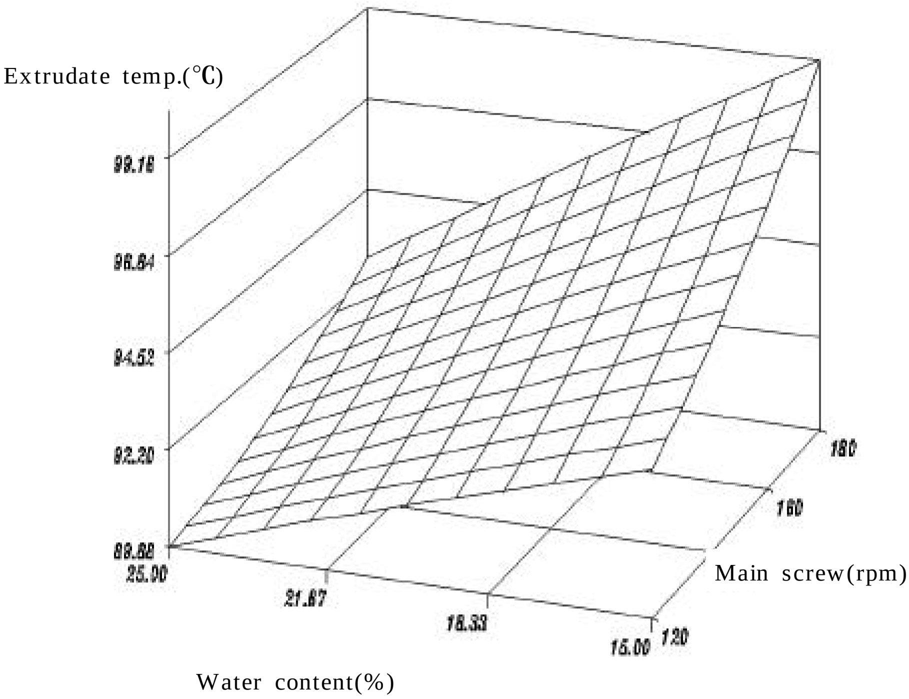


Fig. 3- 5. Response surface of extrudates temperature in terms of water content and screw speed at 1, 2 barrel temperature 60, 80℃

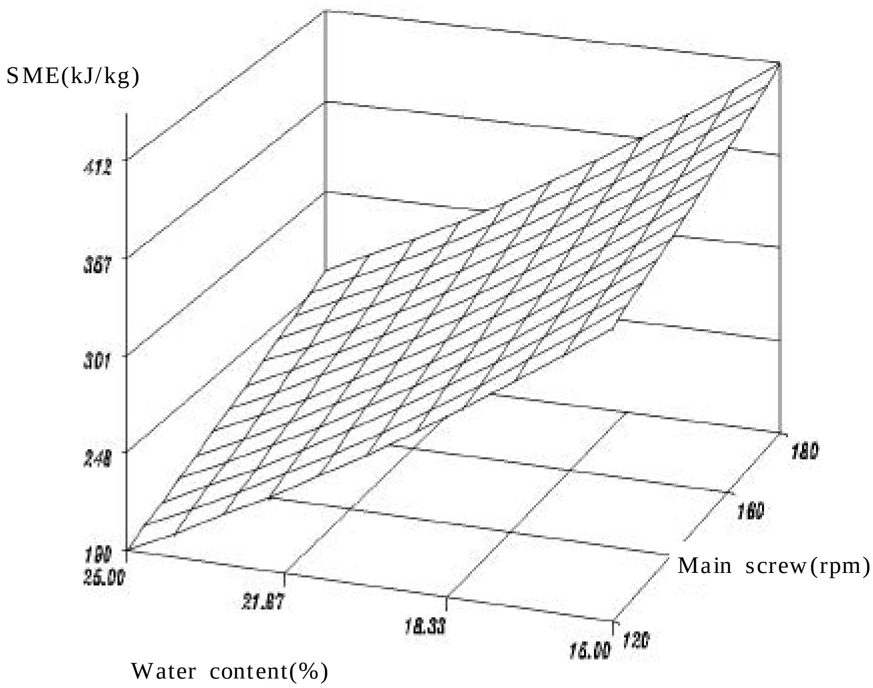


Fig. 3- 6. Response surface of SME in terms of water content and screw speed at 1, 2 barrel temperature 100,120℃

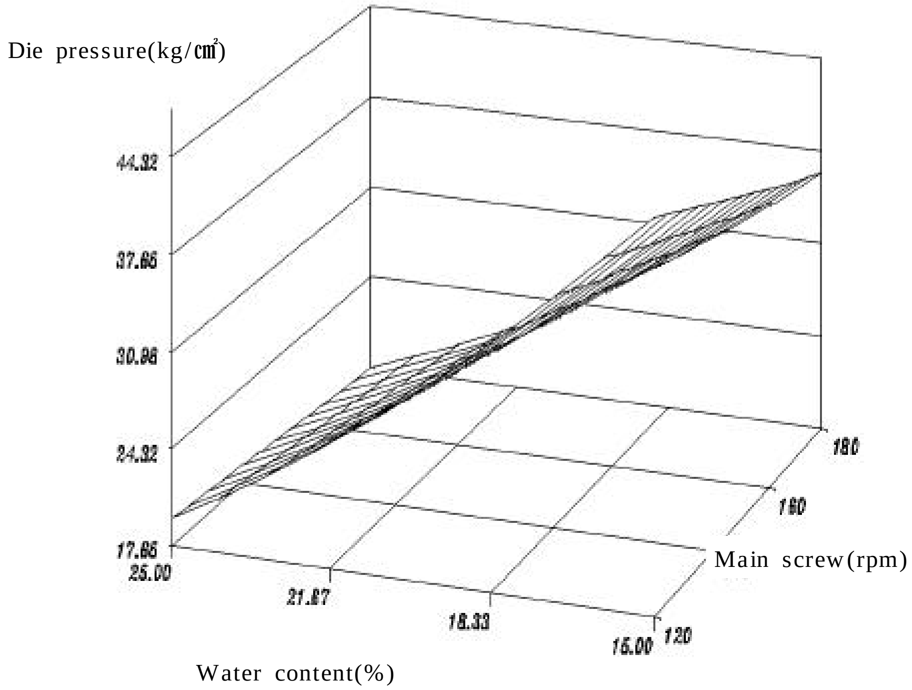


Fig. 3-7. Response surface of die pressure in terms of water content and screw speed at 1, 2 barrel temperature 100, 120℃

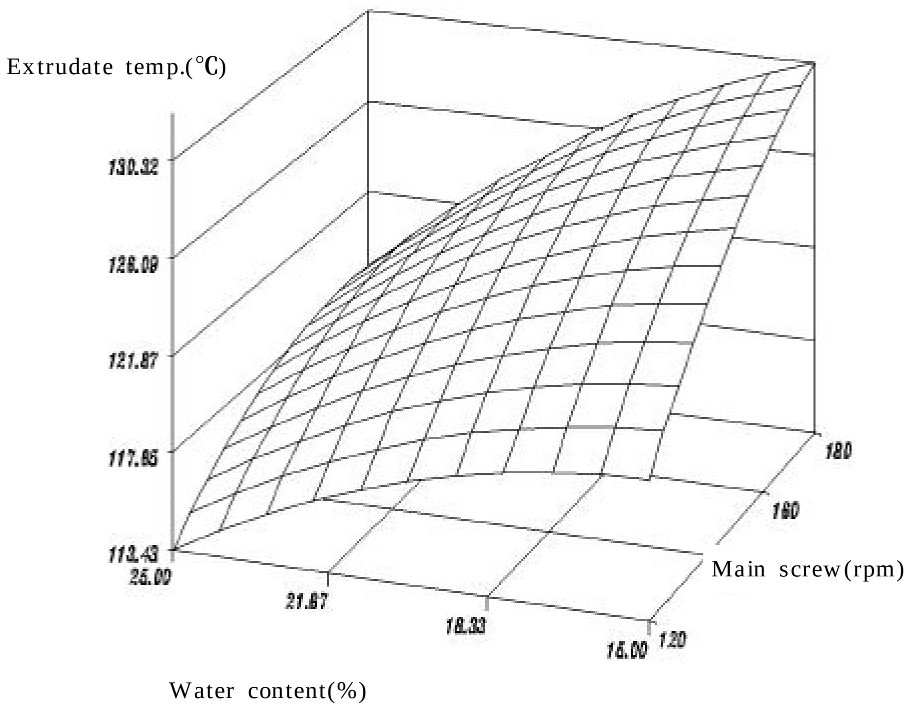


Fig. 3-8. Response surface of extrudates temperature in terms of water content and screw speed at 1, 2 barrel temperature 100, 120℃

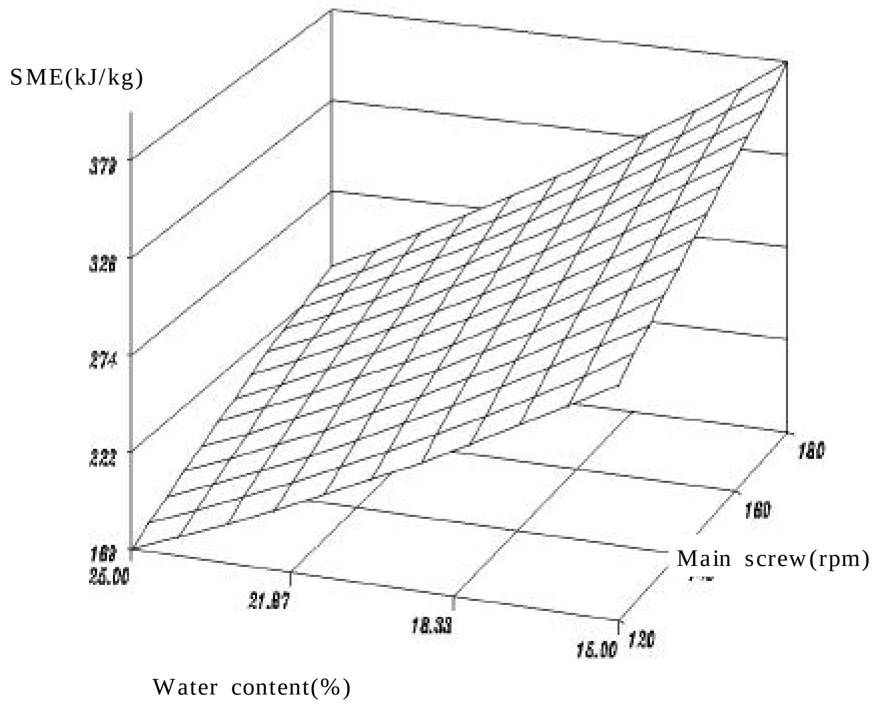


Fig. 3-9. Response surface of SME in terms of water content and screw speed at 1, 2 barrel temperature 140, 160℃

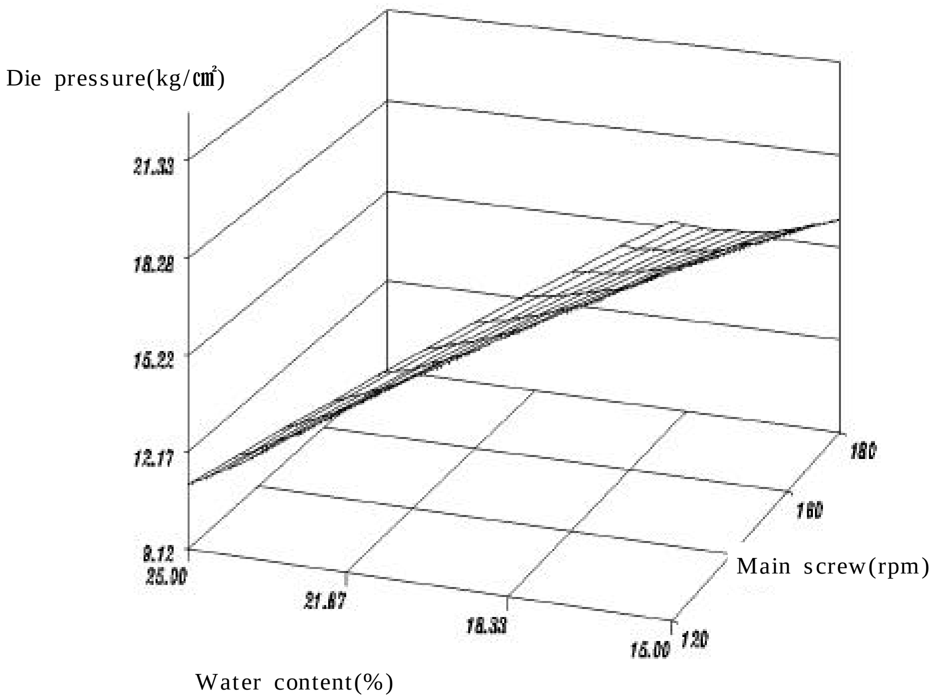


Fig. 3-10. Response surface of die pressure in terms of water content and screw speed at 1, 2 barrel temperature 140, 160℃

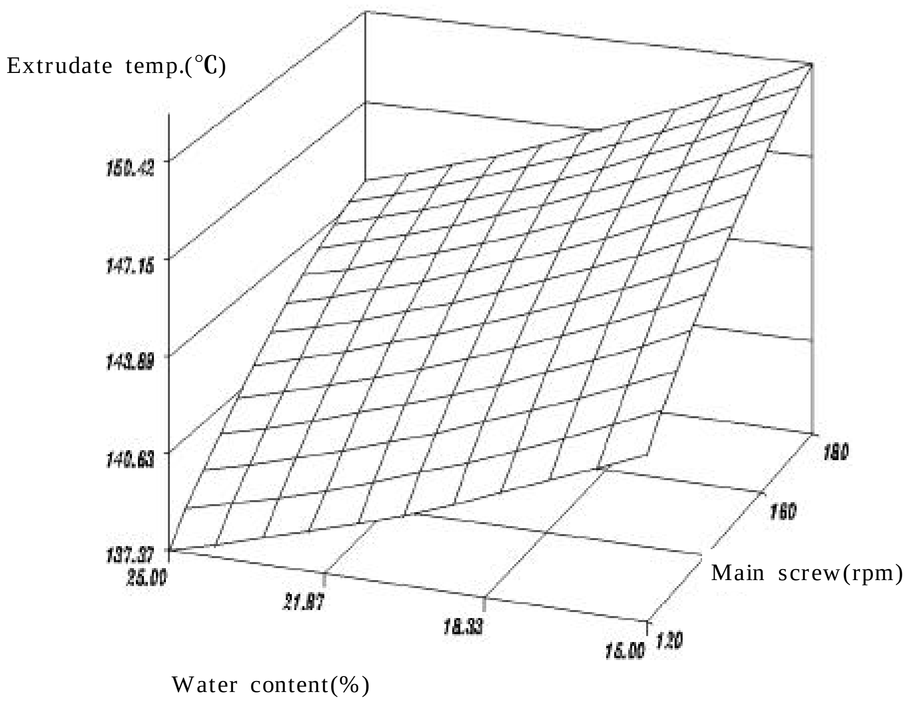


Fig. 3- 11. Response surface of extrudates temperature in terms of water content and screw speed at 1, 2 barrel temperature 140, 160℃

Table 3-1. Dry expanded petfood formulation

Ingredients	**Composition(%)**
Corngrits	62.0
SoybeanOil cake	17.6
Wheat bran	5.0
Bone meal	2.4
Beef tallow	2.5
Animal tallow cake	1.5
Rice embryo	5.0
Calcium carbonate	2.0
Cacium phosphate	0.73
Sodium chloride	0.5
Yeast	0.45
Vitamin mix	0.1
Mineral mix	0.1
Anti-oxidant	0.02
Total	**100.00**

Table 3-3. Extruder operation condition for the preliminary test

Processing variables		
Barrel temp.1) (℃)	Water content (%)	Main screw (rpm)
60-80	15	120
		150
		180
	20	120
		150
		180
	25	120
		150
		180
100-120	15	120
		150
		180
	20	120
		150
		180
	25	120
		150
		180
140-160	15	120
		150
		180
	20	120
		150
		180
	25	120
		150
		180

1) Barrel has two sets of heaters. The first heater has lower temperature than the second one.

Table 3-4. Relationship between process variables, system variables, and physical properties of extrudate in pet food extrusion

Processing variables			System variables			Physical properties		
Barrel temp. (℃)	Water content (%)	Main screw (rpm)	SME[1] (kJ/kg)	Die pressure (kg/cm²)	Extrudate temp. (℃)	E.R.[2]	B.D.[3] (g/ℓ)	W.A.[4]
60-80	15	120	414.01	46.90	93.35	1.225	530.51	68.37
		150	434.91	42.40	99.90	1.252	431.47	78.29
		180	464.49	35.65	106.70	1.304	407.26	91.40
	20	120	287.56	37.10	91.60	1.088	670.62	43.26
		150	334.49	27.55	92.80	1.208	579.30	59.36
		180	351.20	19.95	95.25	1.228	522.10	73.76
	25	120	227.36	20.85	89.10	1.042	720.30	29.91
		150	255.86	17.70	92.55	1.062	695.48	35.51
		180	271.47	15.95	95.15	1.102	651.33	51.56
100-120	15	120	357.00	44.70	118.65	1.274	409.83	85.31
		150	377.63	40.05	129.80	1.321	391.34	91.53
		180	412.01	32.60	140.40	1.323	381.21	92.40
	20	120	258.66	28.70	114.40	1.166	595.58	58.36
		150	294.03	21.80	119.60	1.223	541.14	66.20
		180	326.14	18.70	126.70	1.238	518.14	75.42
	25	120	191.96	16.40	93.25	1.053	706.02	34.75
		150	219.70	14.80	116.15	1.062	689.24	42.99
		180	250.68	12.85	126.00	1.109	633.45	53.19
140-160	15	120	294.10	35.85	146.15	1.342	370.59	95.12
		150	340.57	29.35	147.60	1.342	370.40	98.85
		180	375.62	25.75	150.75	1.351	348.48	99.82
	20	120	220.96	25.75	141.50	1.219	557.10	65.15
		150	262.74	18.45	144.35	1.243	494.23	77.62
		180	302.26	17.30	144.80	1.269	418.90	82.30
	25	120	171.99	11.15	124.20	1.100	653.50	45.19
		150	199.97	9.95	137.10	1.125	609.27	55.01
		180	228.83	8.90	142.45	1.211	651.57	60.46

1) Specific mechanical energy.
2) Expantion ratio
3) Bulk density
4) Water absorption

Table 3-5. Correlation coefficient between process variables, system variables and physical properties of the extrudates

	Barrel temp.	Water content	Main screw speed	SME	Extrudate temp.	Die pressure
Expansion ratio	0.3251	- 0.8734	0.2920	0.8113	0.7081	0.6242
Bulk density	- 0.3283	0.8799	- 0.2979	- 0.8303	- 0.7045	- 0.6442
Water absorption	0.3265	- 0.8654	0.3415	0.8194	0.7238	0.6069

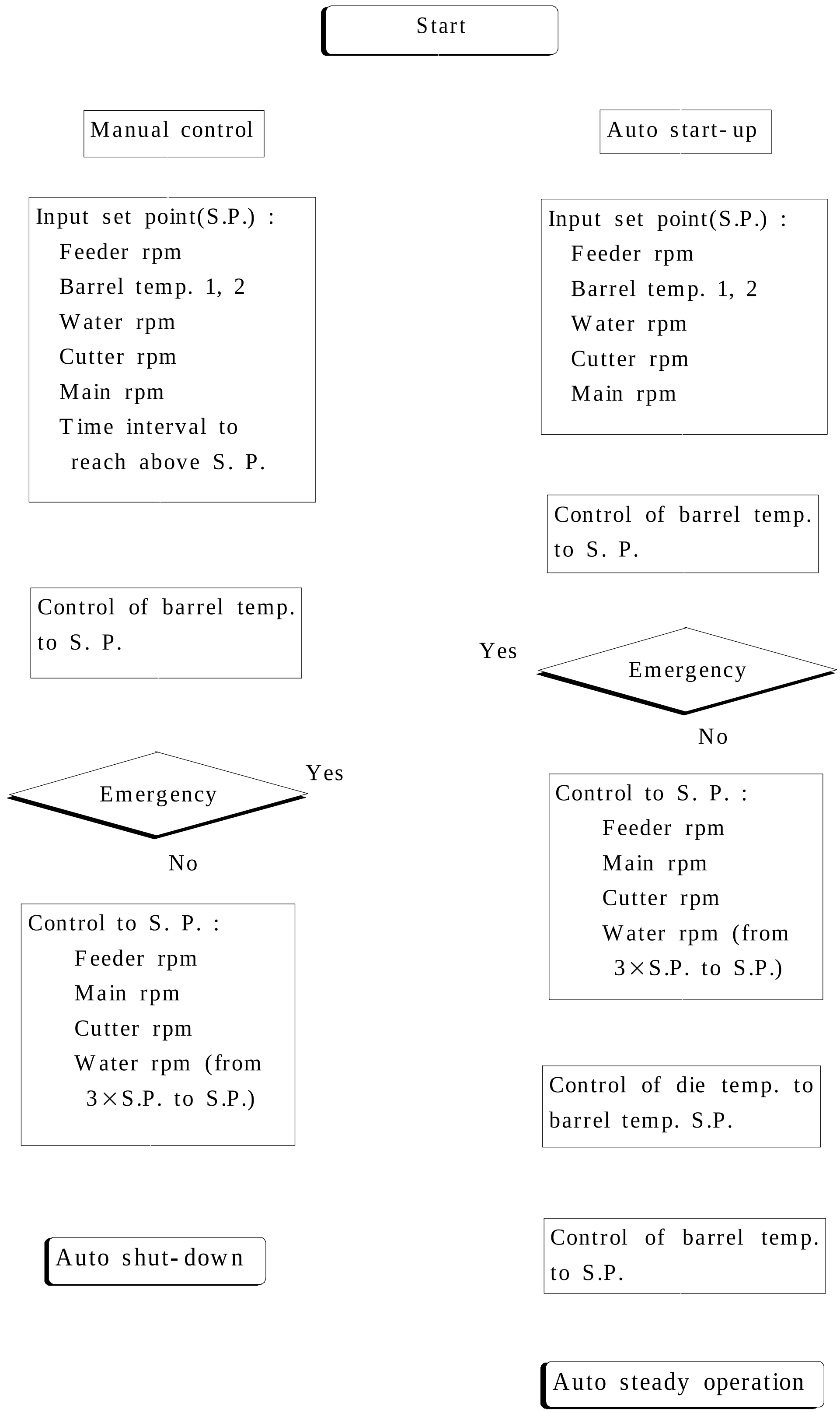
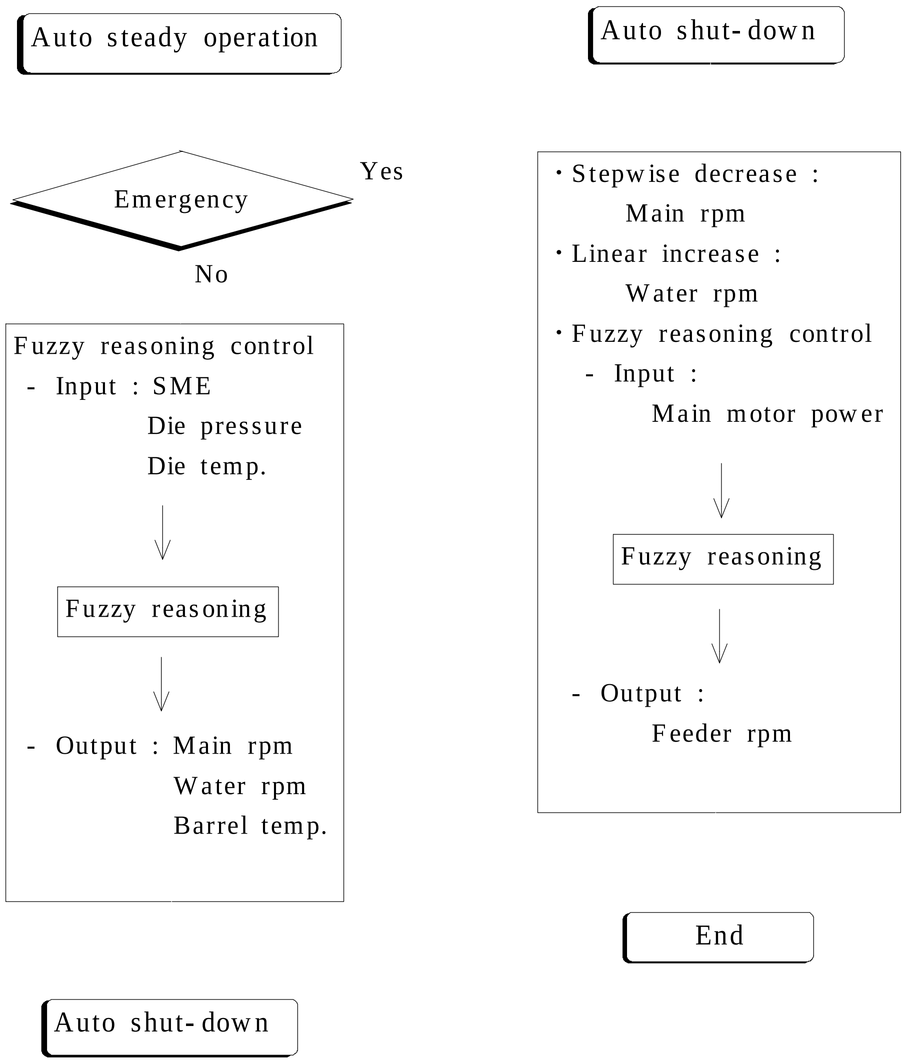


Fig. 1. Control flow chart of manual operation, automatic start-up, automatic steady-operation and automatic shut-down of animal feed extruder


Auto steady operation
Emergency
Yes
No
Fuzzy reasoning control
- Input : SME
Die pressure
Die temp.
Fuzzy reasoning
- Output : Main rpm
Water rpm
Barrel temp.
Auto shut-down
Auto shut-down
· Stepwise decrease :
Main rpm
· Linear increase :
Water rpm
· Fuzzy reasoning control
- Input :
Main motor power
Fuzzy reasoning
- Output :
Feeder rpm
End

(Continued)

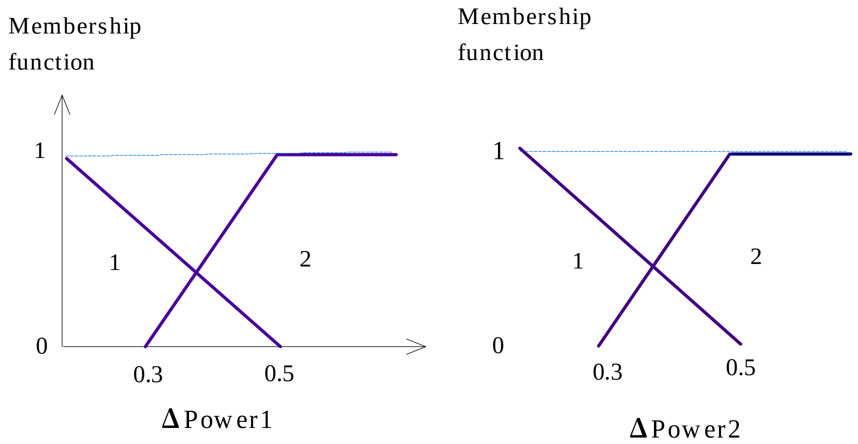


Fig. 3-28. Fuzzy value of gradients 1, 2 of main motor power.
(ΔPower1=Power deference between 3sec,
ΔPower2=Power deference between 6sec)

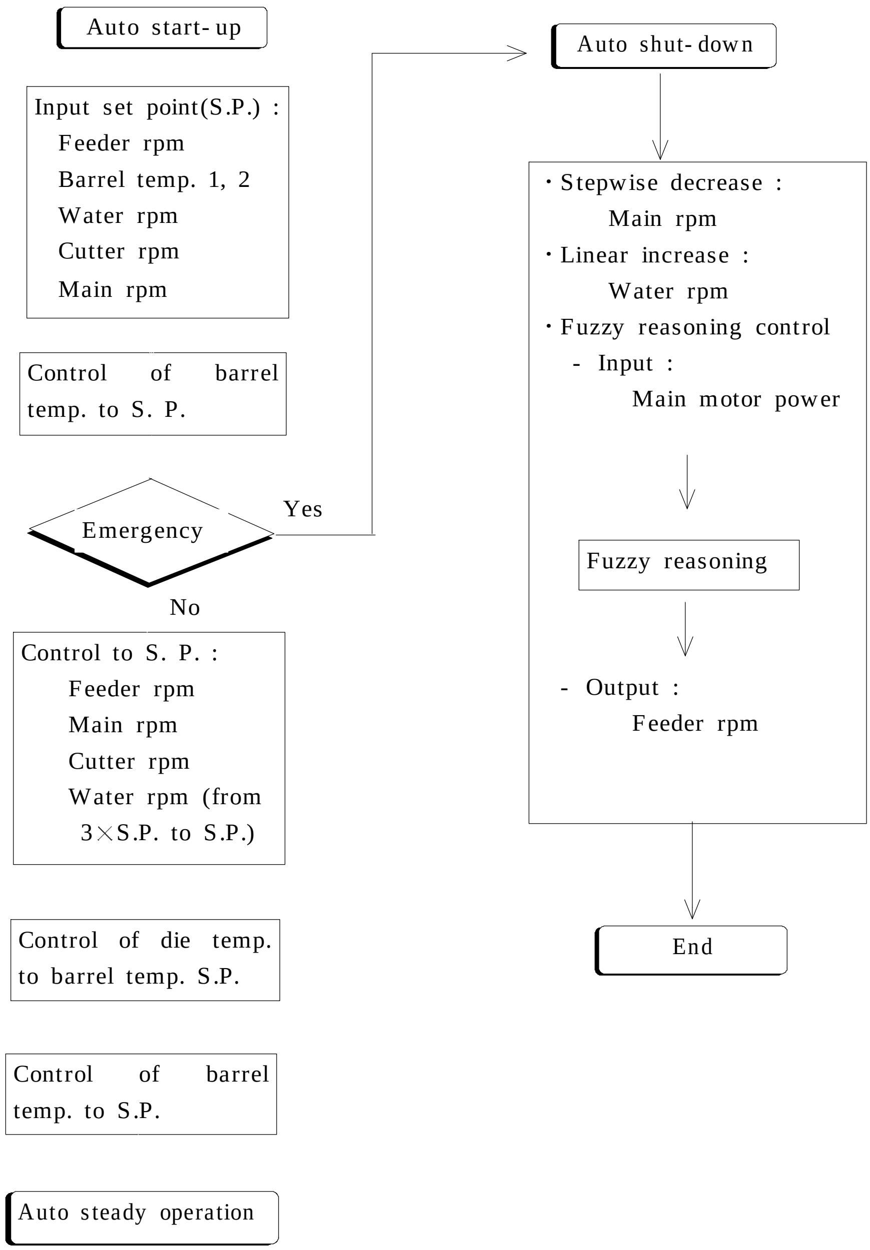


Fig. 3-22. Flow chart of computer program for automatic control of start-up and shut-down.

Table 3-9. Fuzzy expert knowledge of shut-down of extruder

If	ΔPower1	ΔPower2	Then	Reduce feeder rpm
	1	1		5
	1	2		by pass
	2	1		5
	2	2		0

1) ΔPower1 is Power deference between 3sec interval.

2) ΔPower2 is Power deference between 6sec interval.