



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시, 귀하는 원저작자를 표시하여야 합니다.



비영리, 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지, 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

工學碩士 學位論文

전류정보를 이용한 가변속냉동시스템의
열교환기 실시간 고장진단

**On-line Fault Detection and Diagnosis for Heat Exchanger of
Variable Speed Refrigeration System Based on Current Information**



2008年 2月

釜慶大學校 大學院

冷凍空調工學科

李 東 奎

工學碩士 學位論文

전류정보를 이용한 가변속냉동시스템의

열교환기 실시간 고장진단

**On-line Fault Detection and Diagnosis for Heat Exchanger of
Variable Speed Refrigeration System Based on Current Information**

指導教授 鄭 碩 權

이 論文을 工學碩士 學位論文으로 提出함

2008年 2月

釜慶大學校 大學院

冷凍空調工學科

李 東 奎

李 東 奎의 工學碩士 學位論文을 認准함

2008年 2月 23日

主 審 工學博士 尹 政 仁 ㉠

委 員 工學博士 金 鍾 秀 ㉠

委 員 工學博士 鄭 碩 權 ㉠

목 차

Abstract	iii
List of tables	v
List of photographs	v
List of figures	vi
Nomenclature	viii
제 1 장 서 론	1
1.1 연구 배경	1
1.2 연구 내용	3
제 2 장 고장진단의 개요	8
2.1 고장진단 관련 용어	8
2.2 고장진단시스템의 구성	9
제 3 장 전류정보를 이용한 고장진단시스템	12
3.1 전류, 온도, 압력정보를 이용한 고장진단시스템	12
3.1.1 무고장모델[NFM(1)]	12
3.1.2 가변속냉동시스템과 고장진단시스템의 구성(1)	16
3.1.3 고장진단 알고리즘(1)	19
3.2 전류정보만을 이용한 고장진단시스템	23
3.2.1 무고장모델[NFM(2)]	23

3.2.2 가변속냉동시스템과 고장진단시스템의 구성(2)	28
3.2.3 고장진단 알고리즘(2)	31
제 4 장 실험 결과	34
4.1 실험 장치	34
4.2 실험 결과(1)	36
4.3 가변속냉동시스템의 고장진단을 위한 전류정보의 유용성	43
4.4 실험 결과(2)	48
4.5 p-h선도를 이용한 열교환기 고장시의 전류 변화 해석	54
제 5 장 결 론	58
참고문헌	60
학술대회 발표 논문 List	62
Appendix	64
A 선형회귀분석(Linear regression analysis)	64
B 퍼지제어기(Fuzzy controller)	66
C 룩업테이블(Look up table) 생성기 소스(Visual basic)	68
D 퍼지제어기 PLC 소스(Ladder diagram)	73
E 고장진단 소스(Labview)	76
감사의 글	82

On-line Fault Detection and Diagnosis for Heat Exchanger of Variable Speed Refrigeration System Based on Current Information

Lee Dong-Gyu

Department of Refrigeration and Air-conditioning Engineering
Graduate School, Pukyong National University

Abstract

Recently, owing to system integration and automation, refrigeration system has become more complicated one. The refrigeration system is basically consisted of an evaporator, a condenser, a compressor, an expansion valve, and pipes connected to each part. For saving energy and improving efficiency, the use of a variable speed refrigeration system(VSRS)with a compressor driven an inverter and a motor is gradually increasing in the fields of heating, ventilating, and air-conditioning(HVAC) industry. However this tendency also causes difficulty in finding fault causes when the system is failed. Especially, if faults are not detected immediately and are not treated adequately, even though simple faults of the system, they can cause economically huge damage by inducing whole system shut-down. In addition, energy consumption will be increased. Therefore, an efficient on-line fault detection and diagnosis(FDD) is very important to predict the performance degradation early and to prevent faults in advance¹⁾.

The conventional studies for FDD have mainly depended on pressure and temperature information. The pressure and temperature information are basically monitored to confirm safety of the refrigeration system. Moreover, the pressure and temperature reflect static characteristics of a refrigeration system. However they are no longer enough for FDD of a VSRS. Because feedback control which is imperative to a VSRS causes compensation effect on the pressure and temperature.

Inverters are essential to the VSRS in order to control compressor speed. Current from an inverter is useful for FDD of mechanical faults as well as electrical faults. The current information responds sensitively according to thermal load variation of a VSRS in case of mechanical faults. In addition, the current is hardly influenced by the compensation effect of feedback control. Thus, FDD for a VSRS based on the current information is faster and more precise than the one based on the pressure and temperature information. Also the current information can be utilized without extra cost because it is always measured on purpose to prevent overcurrent in an inverter.

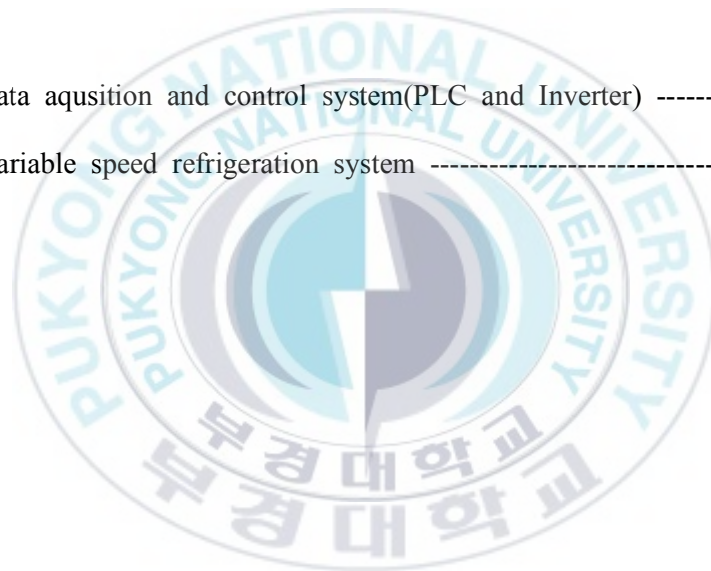
In this paper, two no fault model(NFM) is presented. One used current, pressure and temperature as variables of NFM(1). The other used only current as variable of NFM(2). And each FDD system according to two NFM is indicated such as experimental equipment and algorithm of FDD. Next, the experimental results by two FDD systems is shown. We could know that one by NFM(2) is better than the other by NFM(1) because it was faster and more precise FDD system in case of FDD system by NFM(1). And one by NFM(2) had easily reliability than the other. And the used reason of current in FDD system in VSRS is indicated and the principle of current as variable of FDD system in VSRS is presented.

List of tables

Table 1	Data for regression analysis of NFM(1) -----	13
Table 2	Data of I_{rms} according to f , VO for NFM(2) -----	25
Table 3	Specification of the experimental unit -----	34
Table 4	Time interval of fault imitation and fault ratio at the time interval -----	36

List of photographs

Photo. 1	Data aquisition and control system(PLC and Inverter) -----	35
Photo. 2	Variable speed refrigeration system -----	35



List of figures

Fig. 1	Software elements of an FDD system -----	11
Fig. 2	Composition of FDD Module -----	11
Fig. 3	Composition of Fault Detection part -----	11
Fig. 4	Comparison result between detected current and estimated one -----	15
Fig. 5	Composition of experimental equipment for FDD in VSRS --	18
Fig. 6	Algorithm of fault detection -----	19
Fig. 7	Algorithm of fault identification -----	21
Fig. 8	The mean residual according to reduction of heat exchange area in condenser -----	22
Fig. 9	Tendency of data for NFM -----	26
	(a) f and VO for NFM -----	26
	(b) Real current for NFM -----	26
Fig. 10	Composition of experimental equipment for FDD in VSRS --	30
Fig. 11	Schematic diagram of fault detection and diagnosis -----	33
Fig. 12	Result of on-line FDD about condenser fouling in a VSRS	40
	(a) Current information according to change of fault ratio ---	38
	(b) Residual information according to fault ratio -----	39
	(c) Pressure information according to fault ratio -----	39
	(d) Temperature information according to fault ratio -----	40
	(e) Result of fault identification according to fault ratio -----	40
Fig. 13	Result right after fault imitation -----	42
	(a) Residual of current right after fault imitation -----	42

	(b) Pressure and temperature right after fault imitation -----	42
Fig. 14	The result according to condenser fouling -----	47
	(a) f , VO -----	46
	(b) Condenser temperature -----	46
	(c) Discharged pressure -----	47
	(d) Residual -----	47
Fig. 15	On-line FDD result of VSRS corresponding to condenser fouling -----	50
	(a) f , VO -----	49
	(b) Residual -----	49
	(c) Result of fault isolation -----	50
	(d) Result of identification -----	50
Fig. 16	On-line FDD result of VSRS corresponding to evaporator fan fouling -----	53
	(a) f , VO -----	52
	(b) Residual -----	52
	(c) Result of fault isolation -----	53
	(d) Result of fault identification -----	53
Fig. 17	p-h diagram of basic refrigeration system -----	54
Fig. 18	Change of p-h diagram according to condenser fouling -----	56
Fig. 19	Change of p-h diagram according to evaporator fan fouling	57

Nomenclature

VSRs	가변속냉동시스템(Variable Speed Refrigeration System)	
NFM	무고장모델(No Fault Model)	
FDDS	고장진단시스템(Fault Detection and Diagnosis System)	
I_d	실시간 측정 전류(Detected current)	[A]
I_e	무고장모델에 의한 추정 전류(Estimated current)	[A]
I_r	전류의 대표값(Representable current)	[A]
I_{rms}	전류의 실효값(Root mean square of current)	[A]
ΔI_{mean}	평균전류잔차(Mean residual)	[A]
f	인버터 주파수	[Hz]
VO	전자팽창밸브 개도	[%]
T_a	실내온도	[°C]
T_o	실외온도	[°C]
T_c	응축온도	[°C]
T_{SH}	과열도	[°C]
P_{out}	압축기 토출 압력	[bar]
P_{in}	압축기 흡입 압력	[bar]

제 1 장 서 론

1.1 연구 배경

냉동사이클은 기본적으로 증발기와 응축기 그리고 압축기와 팽창밸브 및 배관으로 구성되어 냉동·공조 산업 분야에 오래 전부터 이용되어 오고 있다. 최근에는 에너지 절약과 효율 향상을 위해서 인버터로 구동되는 압축기 모터를 가진 가변속냉동시스템(Variable Speed Refrigeration System, 이하 VSRS)이 점차 증가하고 있다. 또한 실외기 1대에 다수의 실내기를 갖는 멀티에어컨 시스템과 같이 시스템이 복잡해지고, 공조 시스템과 냉동 창고 등이 갈수록 대형화, 자동화가 빠르게 진행되어 해당 시스템의 효율과 성능이 향상되고 있다. 반면에 이러한 경향은 시스템 구성 요소 간의 관계를 복잡하게 만들기 때문에 시스템 일부에서 고장이 발생할 경우, 고장 원인 및 발생 지점 등을 파악하기 어렵다. 특히, 고장이 조기에 발견되지 않은 채 장기간 방치될 경우 이로 인해 전체 시스템 가동을 중단시키는 결과를 초래하여 효과적인 고장 복구 조치가 지연됨으로써 경제적으로 엄청난 손실을 초래할 수 있다. 게다가 시스템 일부에 고장이 발생하면 전체 시스템에 영향을 미쳐서 에너지 사용량이 증가하고 시스템의 신뢰성과 안정성 및 실내 쾌적성이 저하한다. 이들 시스템의 신뢰성과 안정성 및 실내의 쾌적성을 확보하고 에너지를 절약하기

위해서 미리 성능 저하를 적절하게 감지하고 고장 발생시 신속하게 처리 가능한 효율적인 고장검출 및 진단(Fault Detection and Diagnosis, 이하 FDD)은 매우 중요하다.

기존의 냉동시스템은 주로 온도와 압력을 이용하여 열부하에 관한 고장 진단을 하였다. 그러나 에너지 절약 문제가 심각하게 대두되면서 VSRS의 사용이 보편화되고 피드백제어기의 사용이 증가하게 되었다. 피드백제어기는 제어변수가 실내온도나 과열도 등과 같은 온도정보이므로 온도와 압력은 설정값에 추종되려는 보상효과를 가지게 되고, 이로 인해 온도와 압력정보는 이전의 정속형냉동시스템에서보다 고장진단에 유용하지 않게 되었다. VSRS의 고장진단에 온도와 압력정보만을 이용한 연구 사례도 있었지만 좀 더 빠르고 정확한 고장진단을 위해 전류정보의 이용이 제안되어졌다.

윤종수 등은 가변속냉동시스템의 효율적인 고장진단을 위해 전류정보의 이용을 제안하고, 그 활용 가능성을 제시했다.¹⁾ 냉동시스템의 고장진단은 크게 기계적인 고장과 전기적인 고장으로 나눌 수가 있는데 기존의 고장진단은 주로 기계적인 고장에 국한되어 진행되어 왔다. 하지만 피드백제어기의 사용으로 압축기의 운전환경이 열악해짐에 따라 압축기의 고장 발생 가능성이 커졌고, 결과적으로 압축기의 고장진단 역시 중요해졌다. 압축기의 고장은 편심 결함과 같은 기계적인 고장과 압축기 모터에 인가되는 전압의 상(phase) 불평형과 같은 전기적인 고장으로 나눌 수 있

다. 압축기의 전기적인 고장의 경우, 압축기 모터로는 농형유도전동기가 주로 사용되기 때문에 유도전동기의 고장진단 분야에서 이미 연구되었던 전류, 진동, 소음 등을 이용한 다양한 고장진단법들의 적용이 가능하다. 따라서 본 논문에서 제안하는 전류정보를 활용하면 냉동시스템의 전기적인 고장진단은 충분히 가능하다. 하지만 압축기를 포함한 냉동시스템의 기계적인 고장에 대한 전류정보의 적용 사례는 윤종수 등에 의한 초보적인 선행 연구가 거의 전부인 실정이다. 따라서 본 논문에서는 VSRS의 기계적인 고장에 주목하여 선행 연구에서 확인된 가변속냉동시스템의 고장진단을 위한 전류정보의 가능성을 구체화시켜 실시간 고장진단시스템을 구축하고, 고장진단 알고리즘을 구성하여 실험을 통해 그 유효성을 입증하고자 한다.

1.2 연구 내용

VSRS의 고장진단에 대한 전류정보의 이용 가능성은 선행 연구에서 이미 밝혀진 바 있다. 따라서 선행 연구를 바탕으로 한 고장진단시스템(Fault Detection and Diagnosis System, 이하 FDDS)의 구축이 필요하다. FDDS 구축을 위해 고장을 검출하는 고장검출부(fault detection part)와 이 검출부로부터 얻어지는 정보를 이용하여 고장의 위치 식별과 고장의 정

도를 규명하는 고장진단부(fault diagnosis part)를 구성한다.

고장검출부에서의 고장검출은 전류의 잔차(residual)정보를 이용한다. 잔차정보는 실제 시스템의 상태를 반영하는 실시간 검출 전류(I_d)와 고장이 없는 정상적인 상태에서의 다양한 운전 조건 하에서 얻어진 전류모델로부터 생성되는 추정 전류(I_e)의 차로부터 구해진다. 추정 전류는 정상 상태(normal state) 하에서 얻어진 무고장모델(No Fault Model, 이하 NFM)에서 생성된다. NFM은 실험을 위해 제작된 기본내동사이클 장치에 대해 다양한 운전 조건 하에서의 실험을 통해 얻어진 데이터에 기초하여 선형 회귀분석(linear regression analysis)을 통해 구하였다.

본 연구에서는 두 가지 NFM에 대해 검토한다. 첫 번째는 독립변수로 다른 제어변수와 함께 온도와 압력 정보를 포함시킨 경우의 NFM(1)이고, 두 번째는 독립변수에 온도와 압력 정보를 모두 배제시킨 경우의 NFM(2)이다. 이 때 종속변수는 NFM(1), NFM(2) 모두 전류로 설정하였다. 독립변수는 NFM(1)의 경우, 조작량인 인버터 주파수(f)를 포함한 온도와 압력정보 7개로 구성되고, NFM(2)의 경우, 인버터 주파수(f)와 전자 팽창밸브 개도(VO)로 구성되었다.

제안된 두 NFM 가운데 NFM(2)는 온도와 압력에 대한 피드백제어기의 보상효과는 물론 온도와 압력정보가 갖는 큰 시정수(time constant)의 영

향도 배제시킬 수 있다. 뿐만 아니라, 검출해야 할 정보의 수를 줄임으로써 센서에 소요되는 가격, 센서의 트러블 발생, 그리고 검출정보에 노이즈가 수반될 가능성도 최소화시킬 수 있다.

고장진단부는 고장의 위치를 식별하고 고장의 정도(level)를 규명하는 역할을 한다. 고장진단부에서는 고장검출부에서 생성된 잔차정보를 이용하여 고장식별과 고장규명의 역할을 수행하기 때문에 열교환기 고장에 따른 잔차정보의 특성을 분석할 필요성이 있었고, 분석 결과 잔차정보는 두 가지 정보를 가지고 있음을 알 수 있었다. 하나는 잔차정보의 경향(pattern)이었고, 다른 하나는 잔차정보의 크기였다. 잔차정보의 경향은 응축기 오염의 경우 양(+)의 방향으로 증가하는 경향을 나타내었고, 증발기 팬 고장의 경우 음(-)의 방향으로 증가하는 경향을 나타내었다. 그리고 잔차정보의 크기는 응축기 오염의 경우 오염 정도에 비례하여 증가하였고, 증발기 팬 고장의 경우 팬 고장 모사 후 급격히 감소하였다. 이러한 열교환기의 고장에 대한 잔차정보의 두 가지 특성을 이용하여 고장식별과 고장규명을 포함하는 고장진단부의 알고리즘을 생성했다.

본 논문에서 제안한 고장검출과 고장진단 알고리즘의 효용성을 검증하기 위해 VSRS의 고장진단 실기 실험시스템을 구축하였다. 실험시스템은 크게 3부분으로 제어대상인 VSRS, 이 VSRS의 압축기와 전자팽창밸브를 제어하기 위한 제어부 그리고 FDDS로 구성되었다. VSRS로는 단일 냉동

시스템이 사용되었고, 제어부에서는 이 시스템의 실내온도와 과열도를 제어변수(control variable)로 설정하여 퍼지로직(Fuzzy logic)으로 구성된 피드백제어기를 PLC(Programmable Logic Controller)상에서 구현하였다. 또한, 압축기의 농형유도모터와 전자팽창밸브의 스텝모터를 구동하기 위해 주파수 가변장치인 인버터(inverter)와 스텝모터드라이브가 각각 사용되었다. FDDS는 PC(personal computer)와 DAQ(Data Aquisition, NI사)장비를 이용하여 구현되었다.

실험에서 VSRS의 열교환기 고장 모사는 응축기의 경우, 전열면적을 인위적으로 감소시켰고, 증발기의 경우는 증발기 팬을 원하는 시각에 강제로 정지시키는 방법을 취하였다.

FDDS는 일정한 정밀도를 보증하면서 검출정보의 수가 적을수록, 속응성(quick responsibility)과 신뢰성이 좋을수록 그 성능이 우수하다고 볼 수 있다. 따라서 본 연구에서는 기존의 온도와 압력정보를 배제한 NFM을 최종적으로 제안하고자 한다. 제안한 방법의 성능을 비교하기 위해 우선 온도와 압력정보가 포함된 NFM(1)에 대한 실험을 진행하고 순차적으로 NFM(2)에 대한 실험을 행한 후 그 결과들을 상호 비교하여 고찰한다. 마지막으로 열교환기의 각 고장에 대하여 p-h선도를 이용한 개념적인 해석을 첨부함으로써 각 고장 시의 전류변동 원인을 유추할 수 있도록 하였다.

결론부에서는 본 연구에서 얻어진 주요한 실험결과들을 정리하고, 제안한 시스템의 적용 시 얻어질 수 있는 기대효과를 서술함과 동시에 향후의 연구 진행방향에 대해서도 언급하였다.



제 2 장 고장진단의 개요

2.1 고장진단 관련 용어

제2장에서는 고장진단의 구성에 대하여 서술할 것이다. 그러기 위해서 먼저 용어의 정의가 필요하다.

다음은 고장진단의 용어와 간단한 설명이다.

- 고장(fault) : 시스템 특성 또는 파라미터가 규정조건에서 허용편차 이상 벗어남
- 잔차(residual) : 측정치와 공정모델에 의해 계산된 추정치의 편차 또는 그 편차에 근거한 고장 지시변수
- 고장검출(fault detection) : 시스템 내 고장 발생여부의 결정
- 고장식별(fault isolation) : 고장의 종류, 위치 및 검출시간의 결정
- 고장규명(fault identification) : 고장의 크기와 시변 특성의 결정
- 고장진단(fault diagnosis) : 고장식별과 규명을 포함하며 고장의 종류, 위치, 크기와 검출시간을 결정

2.2 고장진단시스템의 구성

Fig. 1은 FDDS의 구성요소를 나타낸다. FDDS는 크게 고장진단(FDD) 모듈과 평가 모듈로 구성되어진다. 고장진단 모듈은 고장발생시 고장을 진단하고, 고장진단의 결과에 따라 조치를 취하거나 사용자에게 알려주는 역할을 한다.

Fig. 2는 고장진단(FDD) 모듈의 구성을 나타낸다. 고장진단 모듈은 고장검출(Fault detection)부와 고장진단(Fault diagnosis)부로 나뉘어진다. 고장검출부는 잔차정보 생성과 고장 발생 유무를 확인하는 역할을 하고, 고장진단부는 고장검출부에서 생성된 정보를 이용하여 발생한 고장에 대해 고장식별과 고장규명의 역할을 한다. 여러 가지 고장 발생시 잔차정보의 경로에 따라 고장의 위치를 검출할 수 있고, 잔차정보의 크기으로써 고장의 정도를 파악할 수 있다. 잔차정보의 양상은 각 고장에 대하여 독립적이고, 잔차정보의 신뢰성이 높아야 고장진단의 안정성을 확보할 수 있다.

Fig. 3은 고장검출의 구성을 나타낸다. 고장검출부의 목적은 고장발생 유무를 확인하고, 고장진단을 위한 잔차정보를 생성하는 것이다. 잔차정보 발생기구로써 고장검출부는 대단히 중요하므로 고장에는 민감하고, 기타 고장, 잡음 및 외란등에는 둔감하도록 구성되어야 한다. 그러기 위해서는 변수를 고장진단의 목적에 맞게 적절히 선택해야 하고, 선택된

정보가 계측될 때 노이즈를 최소화시켜야 한다. 잔차정보를 생성하기 위해서 여러 가지 방법이 사용되고 있지만 본 논문에서는 모델을 기반으로 하는 방법을 사용한다. VSRS의 정상상태(normal state)의 무고장모델을 생성하기 위해 VSRS의 인버터 주파수와 전자팽창밸브 개도에 따른 정상상태(steady state)값들을 수집하고, 그 값들을 선형회귀분석하여 NFM을 생성하였다. 그리하여 NFM에 의한 추정 전류값과 실시간 전류값을 비교하여 잔차정보를 생성하였다.



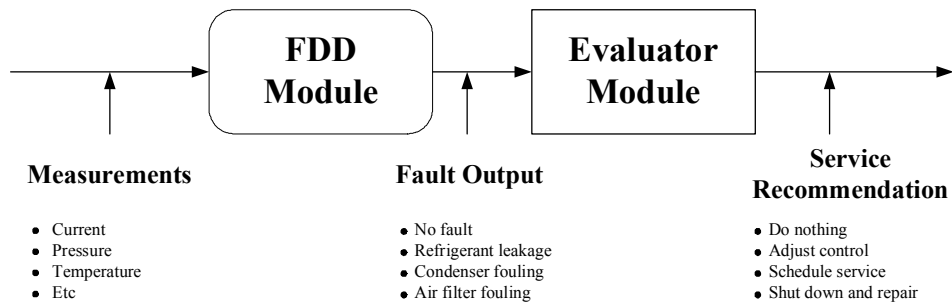


Fig. 1 Software elements of an FDD system

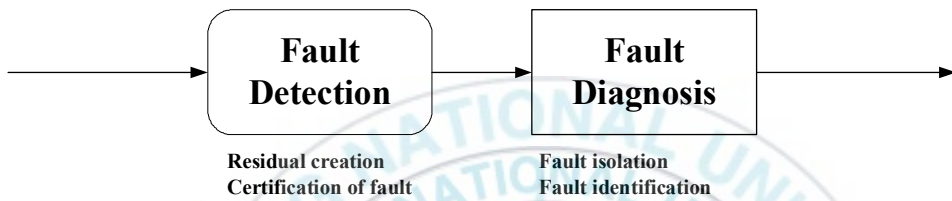


Fig. 2 Composition of FDD Module

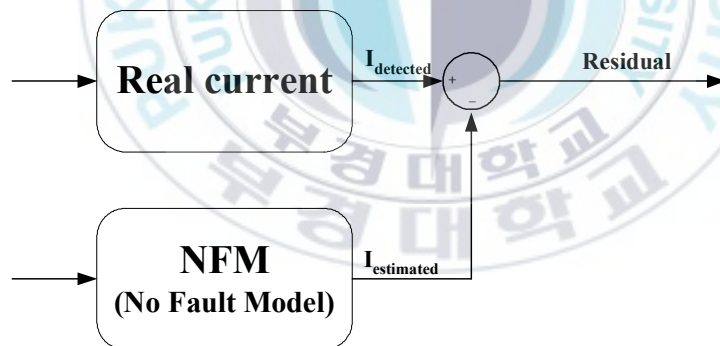


Fig. 3 Composition of Fault Detection part

제 3 장 전류정보를 이용한 고장진단시스템

제3장에서는 실시간 고장진단을 위한 시스템에 대해 기술한다. 고장검출부의 무고장모델에 따라 두 가지 시스템을 소개한다. 첫째로 전류, 온도, 압력을 이용한 고장진단시스템에 대하여 기술한다. 그리고 그에 따른 고장진단 장치도와 알고리즘을 소개한다. 두 번째로 전류정보만을 이용한 고장진단시스템에 대하여 기술하고, 마찬가지로 장치도와 알고리즘을 소개한다.

3.1 전류, 온도, 압력을 이용한 고장진단시스템

3.1.1 무고장모델[NFM(1)]

고장을 진단함에 있어 가장 중요한 것은 고장진단 결과의 신뢰성과 속응성이다. 신뢰성을 좋게 하기 위해서는 속응성이 나빠질 수 있고, 속응성을 좋게 하기 위해서는 신뢰성이 나빠질 수 있다. 하지만 신뢰성과 속응성은 모두 고장진단에 있어서 매우 중요한 요소이기 때문에 한 요소를 포기할 수는 없다. 이 두 마리 토끼를 잡기 위해서는 고장을 검출하기 위한 NFM의 변수 선택이 가장 중요하다. NFM의 변수의 선택은 고장검출의 신뢰성과 속응성뿐만 아니라 전체적인 고장진단의 성능을 좌우하기

때문이다.

먼저 NFM의 변수로써 전류, 인버터 주파수, 온도, 압력을 선정한 경우를 소개한다. 이와 같은 변수들을 선택한 이유는 냉동시스템의 모든 상황을 반영하고자 했기 때문이다. 전류, 인버터 주파수, 온도, 압력에 의한 NFM(이하, NFM(1))은 Table 1의 데이터에 의해 생성되어졌다.

Table 1 Data for regression analysis of NFM(1)

f [Hz]	VO [%]	T_a [°C]	T_o [°C]	T_c [°C]	T_{SH} [°C]	P_{out} [bar]	P_{in} [bar]	I_r [mA]
60	60	-5.5	28	41	5	17.6	2.1	13.7
	55	-5.8	28.5	40.5	7.3	17.3	2	13.4
	50	-5	28.6	40	8.2	17.2	1.9	13.4
	45	-2	28.6	39.2	11.6	17.1	1.95	12.8
	40	1.5	28.9	40.2	15.5	16.9	1.6	12.4
55	80	-2.5	30.3	43.5	1.8	18.5	2.4	8.8
	70	-1.7	30.2	42.6	6	18.2	2.3	8.8
	55	-0.5	29.9	40.7	11.3	17.2	1.9	8.3
50	80	-0.9	29.2	41.3	6.8	17.8	2.5	8.4
	70	-2	28.4	40	8.1	17.2	2.3	9
	65	-1.3	30.2	41.4	7.1	17.6	2.3	8.3
	50	3.8	29.2	39.7	16.5	17	2	8.4
45	100	3.9	30	42.5	7.7	18.1	3	8
	80	5.5	30	41.8	10.8	17.8	2.7	7.8
	60	8.3	29.7	40.5	17.8	17.2	2.5	7.8
40	90	3.5	30	41.7	7.3	17.6	3.15	8.4
	80	11.3	29.9	41.5	15.2	17.3	2.9	8.2
	60	12.5	29.8	39.8	21.5	16.5	2.5	7.3

Table 1은 NFM(1)을 위한 회귀분석 데이터이다. 인버터 주파수와 전자 팽창밸브 개도를 일정하게 유지하고, 정상상태에 도달했을 때 각각의 값들의 대표값을 측정했다. 측정 변수는 $I_r, f, VO, T_a, T_o, T_c, T_{SH}, P_{out}, P_{in}$ 이고, 이 중에서 종속변수는 I_r 이고, 독립변수는 $f, T_a, T_o, T_c, T_{SH}, P_{out}, P_{in}$ 이다. VO 는 다른 독립변수들보다 영향력이 적어 제외시켰다. 선형회귀분석 방법은 Appendix. 1에 첨부하였다.

선형회귀분석 결과 NFM(1)은 식 (1)과 같았다.

$$Y = 21.575 + 0.518X_1 - 0.082X_2 - 1.136X_3 + 1.167X_4 + 0.152X_5 - 3.905X_6 + 5.921X_7 \quad (1)$$

종속변수 Y 는 I_r 를 나타내고, 독립변수 X_1 은 f , X_2 는 T_a , X_3 는 T_o , X_4 는 T_c , X_5 는 T_{SH} , X_6 는 P_{out} , X_7 은 P_{in} 을 나타낸다. 식(1)의 결정계수 R Square는 94.9%로써 매우 높은 적합도를 나타냈다. 따라서 NFM(1)은 시스템의 정상상태를 잘 반영한다는 것을 알 수 있었다.

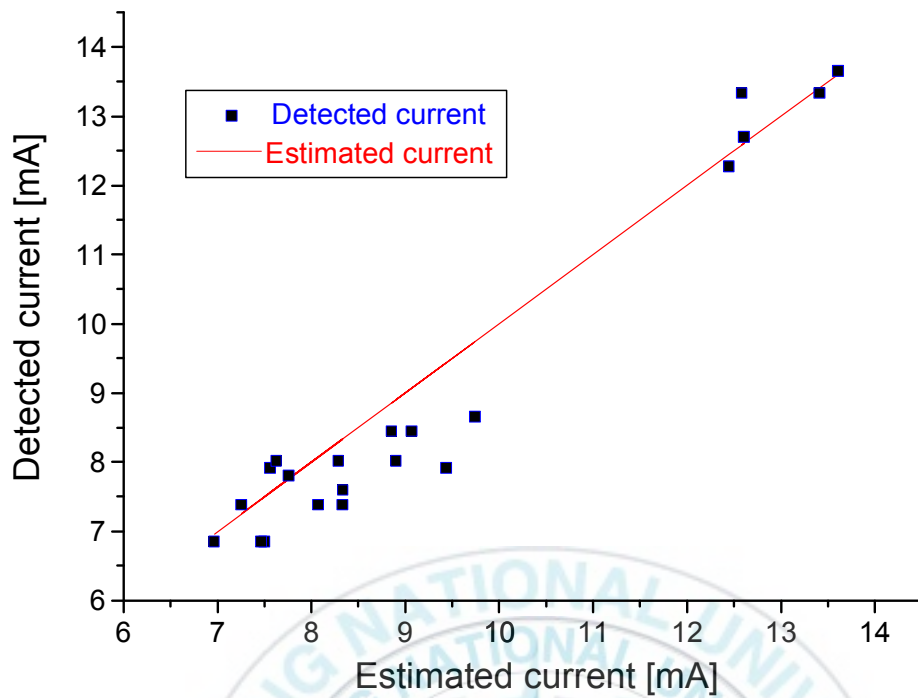


Fig. 4 Comparison result between detected current and estimated one

Fig. 4는 측정된 전류와 NFM(1)에 의해 추정된 전류와의 비교결과를 나타낸 것이다. 측정된 전류가 높은 전류와 낮은 전류로 양분되는 현상은 인버터 주파수와 관련이 있다. 인버터 주파수 60[Hz]일 때는 높은 전류가 측정되었고, 그 외의 주파수에서는 낮은 주파수가 측정되었다.

3.1.2 가변속냉동시스템과 고장진단시스템의 구성(1)

Fig. 5는 전류, 온도, 압력정보를 이용한 가변속냉동시스템의 고장진단을 위한 장치도를 나타낸다. 냉동시스템은 압축기, 응축기, 수액기, 전자팽창밸브, 증발기로 단일 냉동시스템을 구성하였다. 제어를 위해 PLC(Programmable Logic Controller)로써 퍼지로직을 이용한 피드백제어기를 구현하였고, 고장진단을 위하여 NI-DAQ장비로 고장진단에 필요한 정보를 습득하고, Labview로써 프로그래밍하였다. 피드백제어기인 퍼지제어의 설계는 Appendix에 첨부하였다. 피드백제어기는 실내온도와 증발기 입·출구 차인 과열도 각각을 일정하게 유지시키기 위해서 압축기 회전수와 전자팽창밸브 개도를 변화시킨다. 압축기의 회전수를 변화시켜 실내온도를 일정하게 유지시키고, 에너지를 절감시켰다. 그리고 전자팽창밸브 개도를 변화시켜 과열도를 일정하게 유지하므로써 최대의 효율을 내도록 설계되었다. 압축기 회전수를 변화시키기 위한 인버터는 $V/f=const$ 제어 타입을 사용했고, 전자팽창밸브 개도를 변화시키기 위해서 스텝핑 모터 드라이버를 사용했다. 실내온도와 과열도의 설정값은 각각 $3^{\circ}C$ 와 $6^{\circ}C$ 로 용량제어와 과열도제어를 독립적으로 구현하였다. 제어와 고장진단을 위해서 측정된 인자는 전류, 인버터 주파수, 전자팽창밸브 개도, 실내온도, 과열도, 실외온도, 응축온도, 압축기 토출 압력, 압축기 흡입 압력이다. 전류는 인버터와 압축기 사이의 3상 중 한 상에 CT(Current

Transformer)를 연결하여 측정하였다.

고장진단을 위해 전류정보를 계측할 때 인버터의 스위칭방식으로 인해 리플(ripple)이 발생하므로 전류값을 보정해 줄 필요가 있다. 그래서 CT에서 측정된 전류값을 안정화시키기 위하여 평활화 필터를 사용하여 지수 이동평균하였고, 30초당 최대값을 대표값(I_r)으로 지정하였다. 전류의 대표값은 주파수에 따라 그 값이 달라지므로 냉동시스템의 고장유무를 판별하기 적절하다. 인버터 주파수와 전자팽창밸브 개도는 피드백제어기에 의해 결정되므로 그 지령값을 PLC의 D/A Module로부터 NI-DAQ장비로 전송시켰다. 온도정보는 T-type 열전대를 이용하여 측정하였고, PLC의 Thermal Module에 의해 수집되었다. 압력은 고압과 저압 게이지를 사용하여 PLC의 A/D Module에 의해 수집되었다. 온도와 압력정보의 노이즈를 제거하기 위해 각 센서의 위치를 최적으로 선정하고, PLC로 10초당 이동 평균하여 NI-DAQ로 이식시켰으며, Labview로서 평활화 필터를 설계하여 그 정보를 사용하였다. 그 때, 평활화 필터의 지수는 50 또는 100으로 설정하였다.

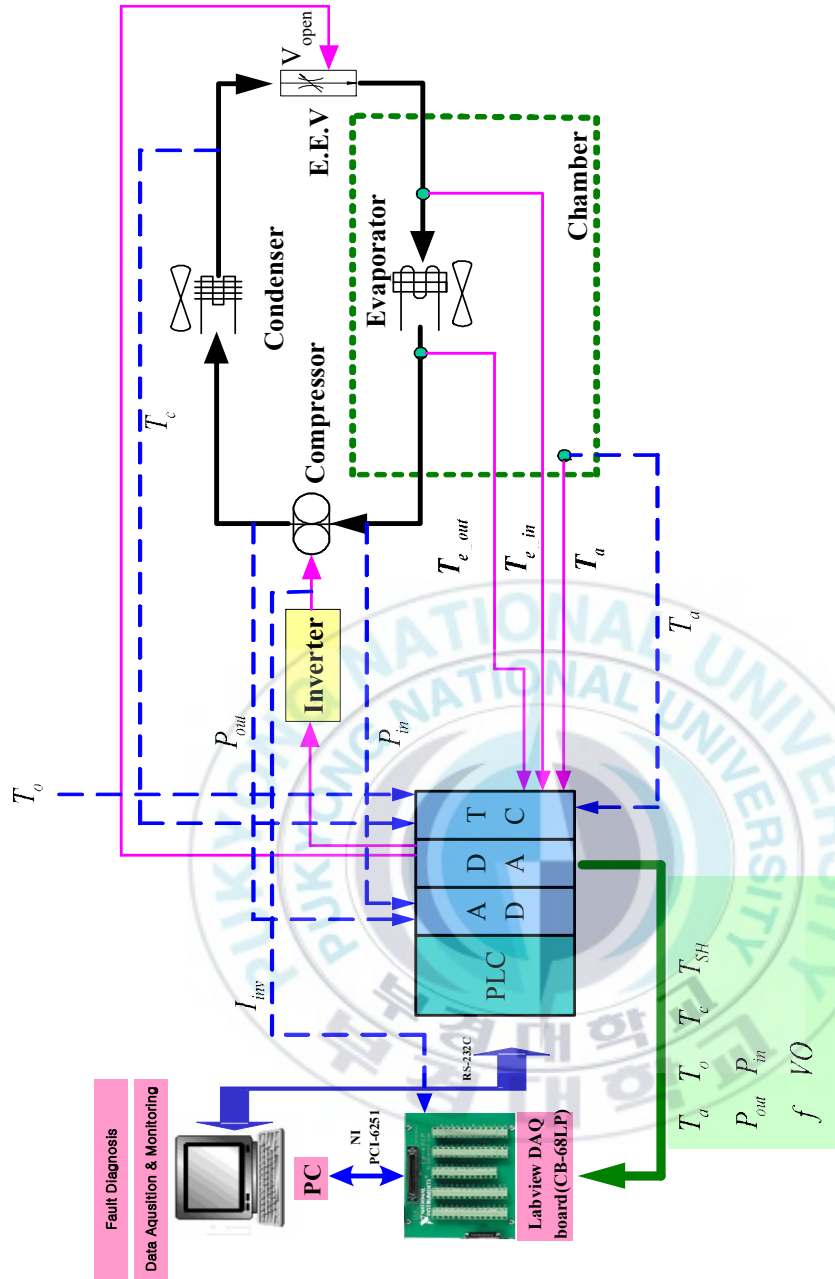


Fig. 5 Composition of experiment equipment for FDD in VSRS

3.1.3 고장진단 알고리즘(1)

이 절에서는 응축기 전열면적에 관한 고장진단에 국한하여 설명한다. 고장진단은 고장검출과 고장진단으로 나뉘어지며 각각의 알고리즘이 필요하다.

Fig. 6은 고장검출에 관한 알고리즘이다. 고장검출부에서는 고장진단의 척도인 잔차정보를 생성하고, 그 잔차정보를 이용하여 고장의 발생 유무를 판단한다. 실시간으로 측정되는 전류 I_d 와 NFM(1)에 의해 추정되는 전류 I_e 의 차가 잔차정보이므로 정상상태에서는 I_d 와 I_e 는 같다. 하지만 응축기 전열 면적이 감소할 수록 I_d 와 I_e 의 차는 커지게 된다. 따라서 잔차정보는 커지게 되고, 그 잔차정보가 고장에 대하여 신뢰성을 가질 수 있는 최소의 크기일 때 고장 발생 유무를 결정짓는 판단 기준이 된다. 고장 발생이 확인되면 잔차정보는 고장진단을 위하여 고장진단부로 전송되게 된다.

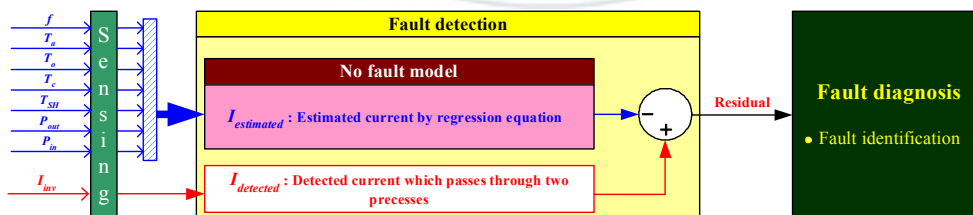


Fig. 6 Algorithm of fault detection

Fig. 7은 응축기 전열면적 감소에 따른 고장규명 알고리즘이다. 고장진단은 고장식별과 고장규명으로 나뉘 수 있지만 응축기 전열면적 감소에 따른 고장만을 다루므로 고장식별은 필요치 않았다. 그래서 이 장에서는 고장규명에 관한 알고리즘만 소개한다.

고장규명을 위해서는 주기가 필요하다. 그 이유는 고장규명의 신뢰성 때문이다. 측정되는 전류 I_d 에는 센서와 측정기기들에 의한 노이즈가 있고, 추정되는 전류 I_e 에도 역시 온도, 압력등의 센서에 의한 노이즈가 있기 때문이다. 고장진단을 위한 판단 주기를 고장진단 주기라 정의한다. Fig. 7의 고장진단 주기는 고장에 의해 시스템이 위험하기까지의 시간의 1/2을 넘어서는 안된다. 본 시스템의 고장진단 주기는 속응성과 신뢰성을 고려한 결과 300초로 설정되었다. 그 300초 사이에 잔차정보가 무고장상태의 범위인 0.001A를 넘는지를 체크하고 0.001A를 넘은 시간이 150초가 넘으면 고장규명을 하게 되고, 그렇지 않으면 정상으로 출력하게 된다. 고장규명의 실시 여부를 위해 카운팅되는 그 시간을 t_{over} 라 정의한다. t_{over} 에 의해 고장규명을 시작하게 되면, 수집되었던 잔차정보를 300초로 나누어 초당 평균값을 산출해 낸다. 그 값을 평균잔차정보(Mean residual, ΔI_{mean})으로 정의한다. 평균잔차정보의 크기에 따라 고장은 규명되고 고장정도가 판명된다.

Fig. 8은 응축전열면적 감소에 따른 평균잔차정보를 나타낸다. 이는 고장규명의 결과인 고장의 정도를 결정하기 위한 데이터이다. NFM(1)을

적용시킨 시스템에 응축전열면적을 감소시킨 후, 그 때의 평균잔차정보를 수집했다. 응축전열면적을 10 ~ 70%까지 감소시켰고, 전열면적 감소량이 증가할수록 평균잔차정보 역시 증가하는 경향을 보였다. 하지만 실험 결과 10%일 때는 평균잔차정보의 신뢰성이 확보되지 않아 고장 검출이 어려울 것으로 판단되었다. 따라서 20~30%에 해당하는 응축기 고장을 Low, 40~50%에 해당하는 응축기 고장을 Middle, 60%이상에 해당하는 응축기 고장을 High로 정의하고 고장규명 알고리즘에 적용시켰다.

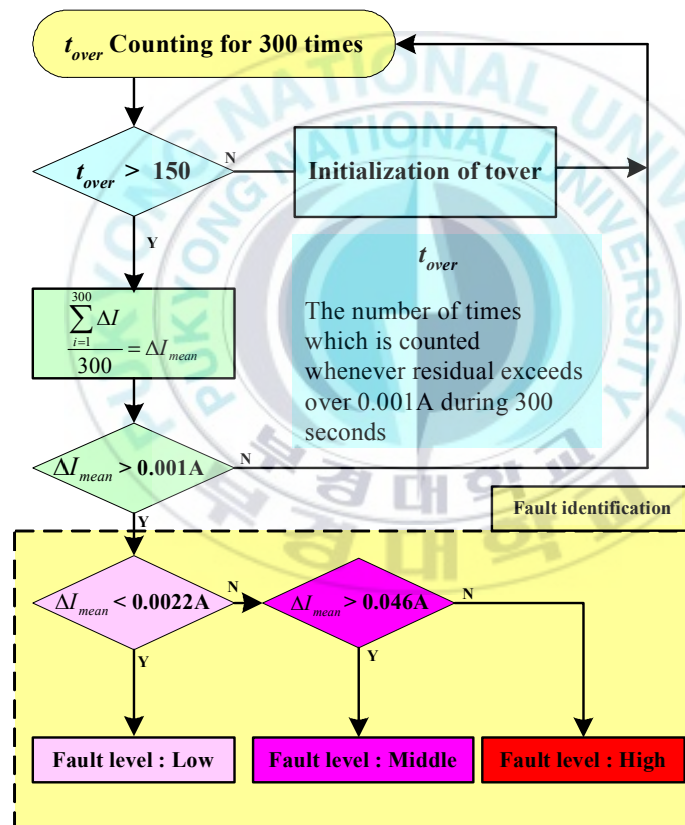


Fig. 7 Algorithm of fault identification

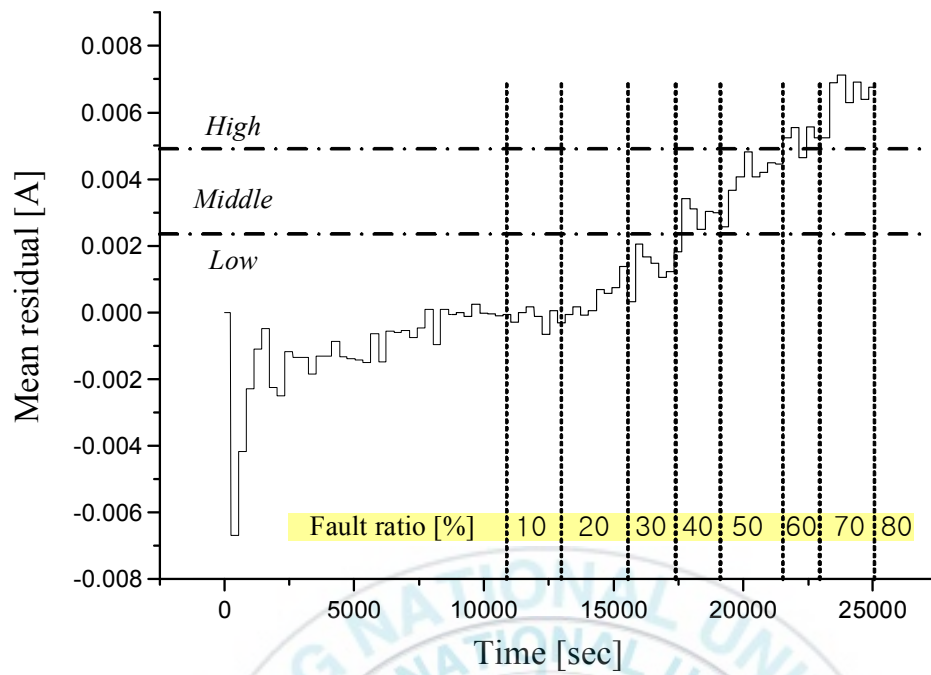


Fig. 8 The mean residual according to reduction of heat exchange area in condenser

3.2 전류정보만을 이용한 고장진단시스템

3.1에서는 전류, 온도, 압력정보를 이용한 FDDS를 소개하였다. 하지만 온도와 압력정보는 피드백제어기에 의한 보상효과와 큰 시정수를 가지기 때문에 FDDS에서 신뢰성을 확보하기 위한 고장진단 주기 역시 길어질 수밖에 없었다. 그리고 다수의 검출 정보로 인해 많은 센서의 수를 동반함으로써 가격이 상승할 뿐만 아니라 센서의 고장과 노이즈가 수반될 가능성이 높아 신뢰성도 떨어질 수 밖에 없었다. 그렇기 때문에 고장진단을 위한 변수로 온도와 압력정보를 배제시키는 방안이 최적의 방안으로 생각되었다. 따라서 본 절에서는 온도와 압력정보를 배제시키고, 전류정보만을 이용한 FDDS를 소개한다.

3.2.1 무고장모델[NFM(2)]

고장 검출부는 실시간으로 측정되는 전류 I_e 와 NFM의 추정된 전류 I_d 를 비교하여 잔차정보를 생성한다. 전류정보만을 이용하여 추정 전류 I_d 를 생성시키기 위한 NFM(이하, NFM(2))에 대해 소개한다.

NFM(2)를 위한 변수를 선정함에 있어 온도와 압력정보를 제외시키고, 전류, 인버터 주파수, 전자팽창밸브 개도를 선택하였다. 피드백제어기에 의한 변화에 가장 빠른 정보를 제공할 수 있는 것은 제어기의 지령값인

인버터 주파수와 전자팽창밸브의 개도이기 때문이다. 온도와 압력정보의 경우는 시정수가 커서 반응이 느리기 때문에 긴급한 고장의 경우 진단이 늦어 시스템에 무리를 주거나 극단적으로는 동작 불능의 상태로 만들 수 있기 때문이다.

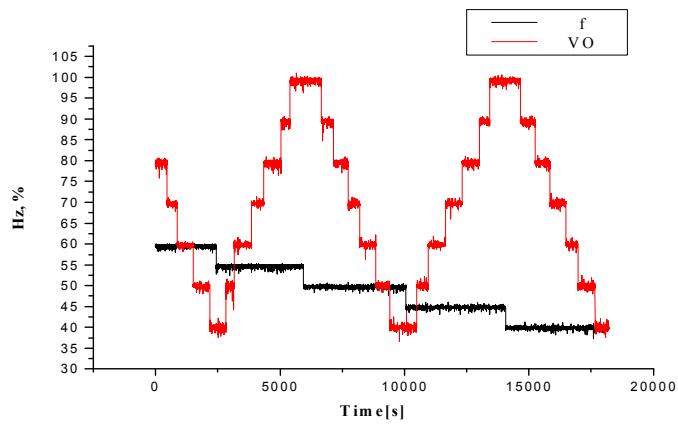
Table 2는 NFM(2) 생성을 위한 I_{rms} , f , VO 의 일부를 나타낸 것이다. 전류정보는 실효값을 사용하였다. 각각 인버터 주파수 f 와 전자팽창밸브 개도 VO 를 고정하고, 충분히 정상상태에 도달했을 때 값을 기록하였다.

Fig. 9 (a)는 NFM(2)을 위한 데이터를 획득하기 위한 f 와 VO 의 변화를 나타낸다. f 와 VO 의 변화는 전류정보의 실효값을 획득하기 위한 지령값으로 f 는 40~60[Hz]의 범위사이에 5[Hz]단위로 변화시켰고, VO 는 각 f 마다 60~100[%]의 범위에 10%단위로 변화시켰다.

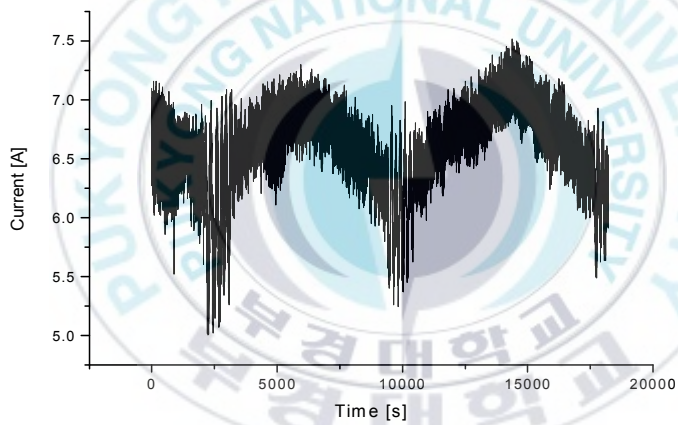
Fig. 9 (b)는 f 와 VO 의 변화에 따른 I_{rms} 의 변화를 나타낸 것이다. I_{rms} 의 경향은 f 보다는 VO 와 비슷했다. 따라서 I_{rms} 는 f 보다는 VO 에 영향을 많이 받을 것이라는 것을 예측할 수 있다.

Table 2 Data of I_{rms} according to f , VO for NFM(2)

f [Hz]	VO [%]	I_{rms} [A]
60	80	6.70
	70	6.62
	60	6.58
	50	6.44
55	100	6.93
	90	6.86
	80	6.75
	70	6.72
	60	6.57
50	100	6.9
	90	6.86
	80	6.72
	70	6.62
	60	6.50
45	100	7.04
	90	6.92
	80	6.78
	70	6.73
40	100	7.13
	90	7.04
	80	6.86
	70	6.75



(a) f and VO for NFM



(b) Real current for NFM

Fig. 9 Tendency of data for NFM

Table 2의 데이터를 이용하여 NFM(2)을 생성하기 위해 회귀분석을 수행하였다. 선형회귀분석 결과 식 (2)의 결과가 도출되었다.

$$Y = 0.683 - 0.007X_1 + 0.01X_2 \quad (2)$$

식 (2)은 NFM(2)을 위한 선형회귀분석의 결과를 나타낸다. 식 (2)의 독립변수 X_1 , X_2 는 각각 f 와 VO 이고, 종속변수 Y 는 I_{rms} 이다. 식 (2)의 R Square는 93.3%으로 높은 적합도를 나타냈다. 따라서 식 (2)은 NFM(2)으로서 적합함을 알 수 있다.



3.2.2 가변속냉동시스템과 고장진단시스템의 구성(2)

Fig. 10는 전류정보를 이용한 가변속냉동시스템의 고장진단을 위한 장치도를 나타낸다. 가변속냉동시스템의 고장진단 장치도는 3.1.2와 내용이 유사하므로 차이점만 설명하도록 한다. 피드백제어기를 위한 설정 실내 온도와 과열도는 11℃와 6℃로 각각 설정하였다. 제어와 고장진단을 위해서 측정된 인자는 전류, 인버터 주파수, 전자팽창밸브 개도이다. 고장진단을 위해 전류정보를 계측할 때 인버터의 스위칭방식으로 인해 리플(ripple)이 발생하므로 전류값을 보정해 줄 필요가 있다. 리플정보는 스위칭과정에서 발생하는 이상(abnormal) 전류이다. 따라서 인버터 주파수 지령값은 한 가지이지만 출력값은 여러 가지 주파수를 포함하게 된다. 이를 제거하는 방법은 하드웨어적인 필터링 방법이 있고, 소프트웨어적인 필터링 방법이 있다. 본 연구에서는 Labview를 이용하여 High pass filter와 Low pass filter를 프로그래밍하여 소프트웨어적으로 최소화시켰다. High pass filter의 경우 38Hz로 설정하였고, Low pass filter의 경우 62Hz로 설정하여 40~60Hz의 전류정보를 안정적으로 받을 수 있게 하였다. 인버터 주파수와 전자팽창밸브 개도는 PLC내의 퍼지로직에 의해 결정되므로, 그 정보를 PLC의 D/A Module에서 NI-DAQ장비로 전송시켰다. 하지만 전송과정에서도 노이즈가 발생하므로 Labview로서 평활화필터를 설계하여 노이즈를 최소화시켰다. 인버터 주파수와 전자팽창밸브 개도의 정

보는 온도와 압력정보와 같은 센서에 의한 값이 아니기 때문에 노이즈가 많지 않아 평활화 필터의 지수는 50으로 설정하였다. 온도와 압력의 측정은 3.1.2의 내용과 동일하다.



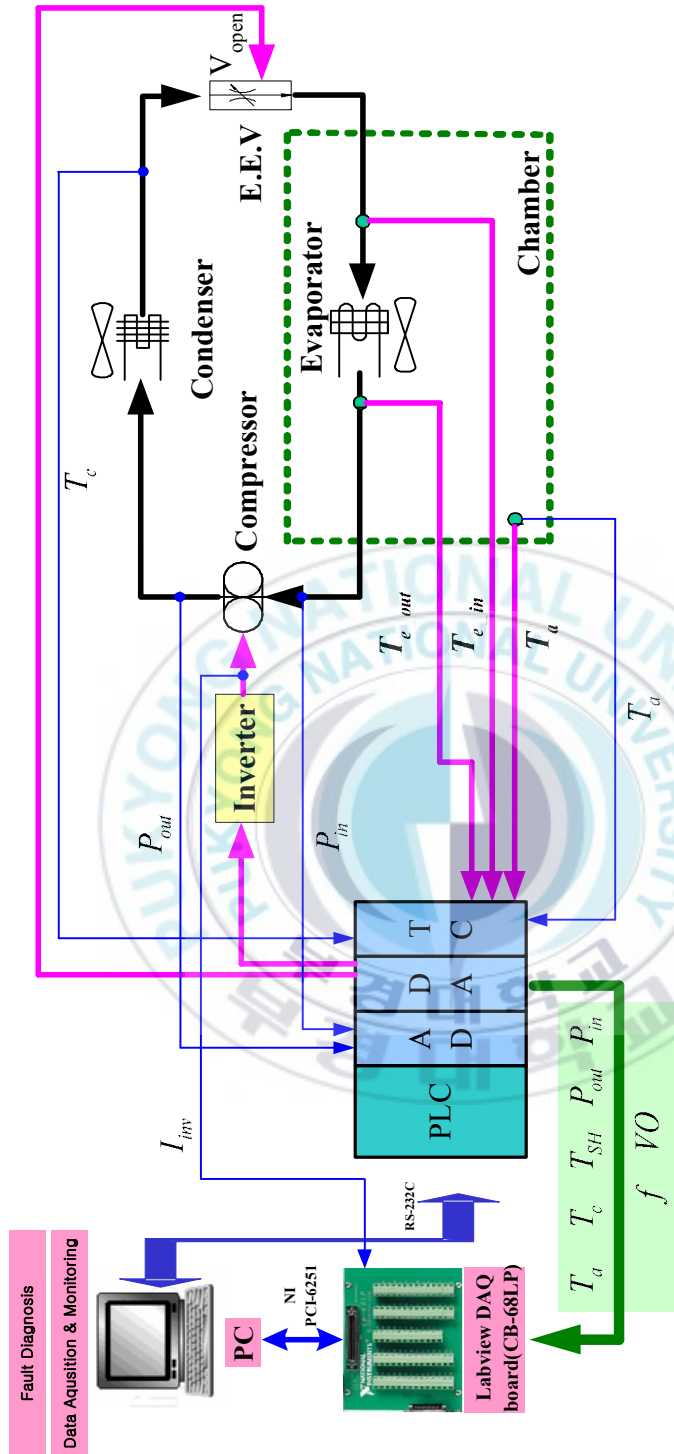


Fig. 10 Composition of experimental equipment for FDD in VSRS

3.2.3 고장진단 알고리즘(2)

Fig. 11은 전류정보를 이용한 가변속냉동시스템의 열교환기 고장진단시스템의 알고리즘을 나타낸다.

고장검출부에서는 잔차정보를 생성하고, 고장의 발생 유무를 판단한다. 잔차정보는 NFM(2)에 의한 추정된 전류 I_e 와 실시간으로 측정되는 인버터 전류 I_d 의 차에 의해서 생성되고, 고장 발생 유무의 판단은 무고장상태의 범위인 잔차정보 $-0.5 \sim 0.2[A]$ 의 범위를 벗어나면 고장 발생 신호를 보내도록 설정하였다.

고장진단부는 고장검출부의 고장 발생 신호에 의해서 동작한다. 잔차정보는 인버터의 리플, 측정 장비의 노이즈, NI-DAQ장치의 노이즈, PLC Module의 노이즈 정보를 모두 담고 있기 때문에 신뢰성을 가지기 위해서 60초 단위로 이동평균하여 고장식별과 고장규명을 할 수 있도록 하였다. 큰 시정수와 피드백제어기에 의한 보상효과를 가진 온도와 압력정보를 배제시킴으로써 NFM(1)에 의한 고장진단시스템의 300초보다 짧은 주기 설정이 가능하였다. 고장이 발생하면 먼저 고장식별을 하게 된다. 고장식별에서는 잔차정보를 초당 잔차정보인 평균잔차정보로 연산하고, 그 평균잔차정보가 상승하면 응축기 고장이고, 감소하면 증발기 고장으로 고장의 위치를 식별도록 했다. 고장식별이 끝나면 고장규명을 하게 된다. 고장의 위치에 따라 다른 알고리즘으로 설계하였다. 응축기 오염의 경우,

평균잔차정보가 $0.2 \sim 0.5[A]$ 의 범위를 가지면 고장의 정도를 30%로 정하고, $0.5[A]$ 이상일 때는 고장의 정도를 50%로 정하여 시스템을 설계하였다. 그리고 증발기 팬 고장의 경우는 빠른 시간내에 시스템에 무리를 주고, 실내온도 상승으로 인한 실내환경이 악화될 수 있으므로 빠른 초치가 필요하다. 따라서 평균잔차정보가 $-0.5[A]$ 이하이면 증발기 팬 고장을 출력하도록 설계하였다.



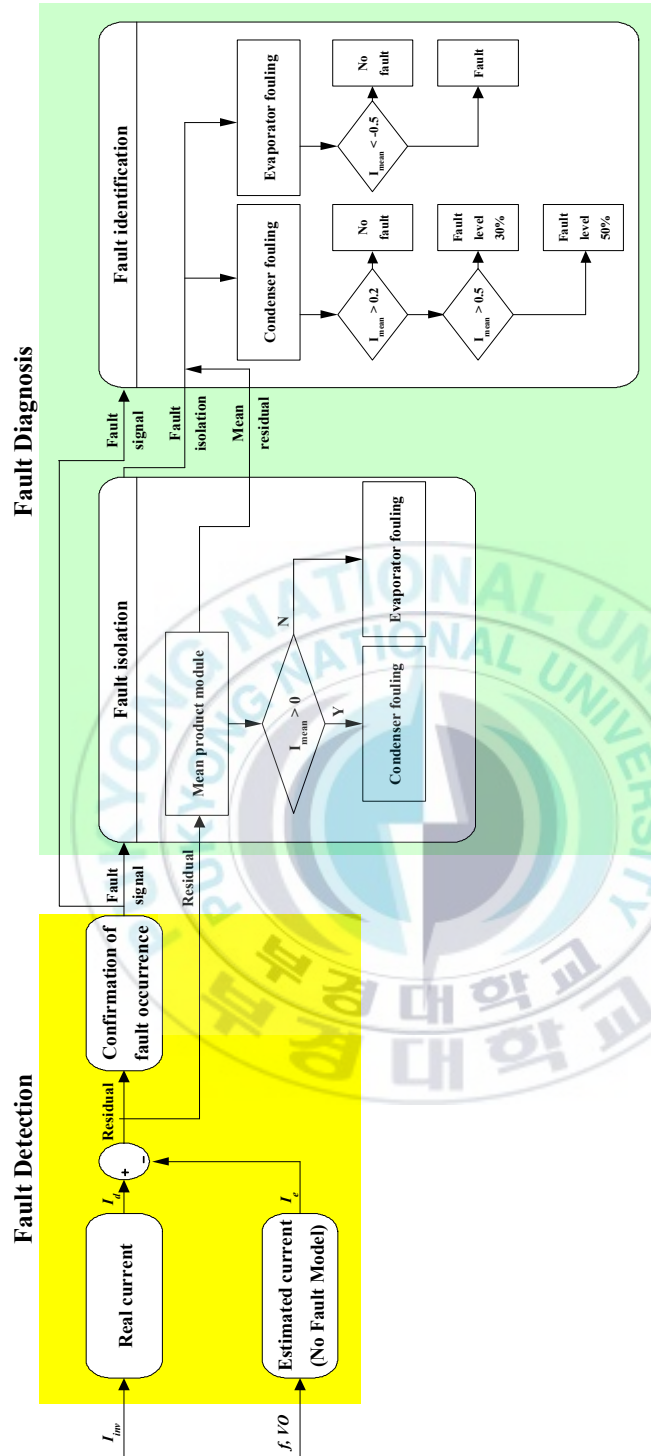


Fig. 11 Schematic diagram of fault detection and diagnosis

제 4 장 실험 결과

4.1 실험 장치

Photo. 1과 Photo. 2는 각각 DAQ(Data Acquisition)과 제어시스템 그리고 가변속냉동시스템의 실제 장치의 모습을 나타낸다.

Table 3는 실험장치의 사양을 나타낸다.

Table 3 Specification of the experimental unit

Compressor	Type Vertical, Reciprocating	Power 220V, 60Hz, 1.5kW
Condenser	Type Fan fin type	
Evaporator	Type Pin-tube type	Capacity 680kcal/h
Expansion valve device	Type EEV Model JHEV 14A Port size $\Phi 14$ Operating range 0 ~ 506 pulse Rated voltage DC 12V	
Refrigerant	Type R22	
Inverter	Type PWM HP 3	
Step valve control interface	Input voltage DC 24V Input signal 4 ~ 20mA Output 0 ~ 400step	
PLC	DC unit 32CH Relay unit 32CH A/D unit 16CH D/A unit 16CH TC unit 16CH CPU GM2	
Chamber	Size 1200×700×1650mm ³	
NI equipment	NI-6251 CB-68LP	



Photo. 1 Data acquisition and control system(PLC and Inverter)



Photo. 2 Variable speed refrigeration system

4.2 실험 결과(1)

NFM(1)을 이용한 VSRS의 FDDS를 이용하여 응축전열면적 감소에 따른 고장진단을 수행하였다.

Table 4는 응축기전열면적을 감소시킨 시간과 감소량을 나타낸다. 응축기 전열면적은 10~70%까지 다양하게 감소시켰고, 각 고장을 일으킨 후에는 시스템이 정상상태로 회복할 시간을 두어 다음 고장에 대하여 영향이 없도록 하였다.

Table 4 Time interval of fault imitation and fault ratio at the time interval

Time[sec]	Fault ratio[%]
1,200 ~ 3,100	10
4,000 ~ 6,000	20
6,500 ~ 8,400	30
9,200 ~ 11,300	50
12,000 ~	70

Fig. 12는 응축기 전열면적 감소에 따른 고장진단 결과이다. (a)는 I_d 와 I_e 를 나타낸다. 고장의 정도에 따라 I_d 는 상승하였고, I_e 는 대폭 감소하는 결과를 보였다. 이는 NFM(1)의 독립변수 중 압축기 토출 압력인 P_{out} 의 영향이 큼을 알 수 있었다. 응축기 전열면적을 감소시키면 증발기 입구 온도가 상승하여 과열도가 감소하게 된다. 따라서 피드백제어기에 의해 팽창밸브의 개도는 감소하게 되고, 냉매량은 줄어들게 되어 냉매의 순환이 원활하지 못하게 된다. 그리고 실내온도는 냉매의 플래시 가스 발생

으로 냉동능력이 감소하므로 상승하려는 경향을 보이고, 그에 따라 압축기 회전수는 증가하는 경향을 보인다. 냉매는 불응축상태에서 팽창되어 증발기로 들어가게 되고, 적은 냉매가 많은 압축일량을 받으므로 압축기 토출압력이 상승하게 된다. NFM(1)에서 압축기 토출압력의 독립변수 X_6 는 음의 부호를 가지므로 (a)와 같이 I_e 가 크게 감소하는 결과를 나타내게 된다.

(b)는 응축전열면적 감소에 따른 잔차정보와 평균잔차정보를 나타낸다. 잔차정보는 (a)의 결과에 의해 생성되고, 응축전열면적 10~20% 감소에서의 경향은 정상상태와 크게 다르지 않다는 것을 알 수 있다. 하지만 응축전열면적 30% 감소부터는 잔차정보의 경향은 정상상태와는 다르게 상승하는 것을 볼 수 있다. 그리고 고장의 정도가 증가할수록 잔차정보의 크기도 증가하는 것을 볼 수 있다. 이에 따라 고장진단 주기로 평균잔차정보도 상승하는 결과를 나타내었다.

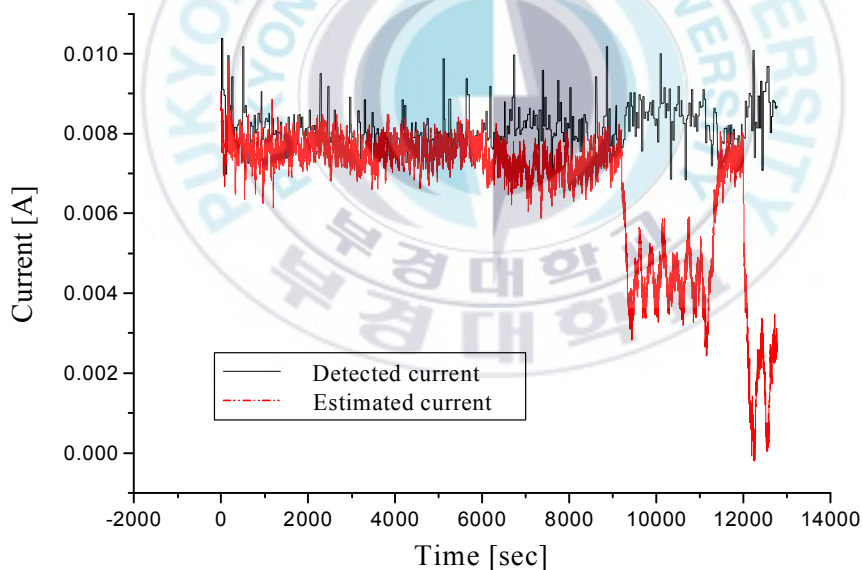
(c)는 응축전열면적 감소에 따른 압력의 변화를 나타낸다. 압축기 흡입압력의 경우 큰 변화를 보이진 않았지만 압축기 토출 압력은 응축기 전열 면적 감소를 잘 반영하는 것을 알 수 있었다.

(d)는 응축전열면적 감소에 따른 온도의 변화를 나타낸다. 피드백제어의 제어변수인 실내온도와 과열도는 고장 후 일정하게 유지하려는 경향을 보이는 것을 알 수 있었다. 따라서 고장진단의 변수로서는 적절하지 않음을 알 수 있었다. 실외온도 역시 일정하게 유지되므로 고장진단

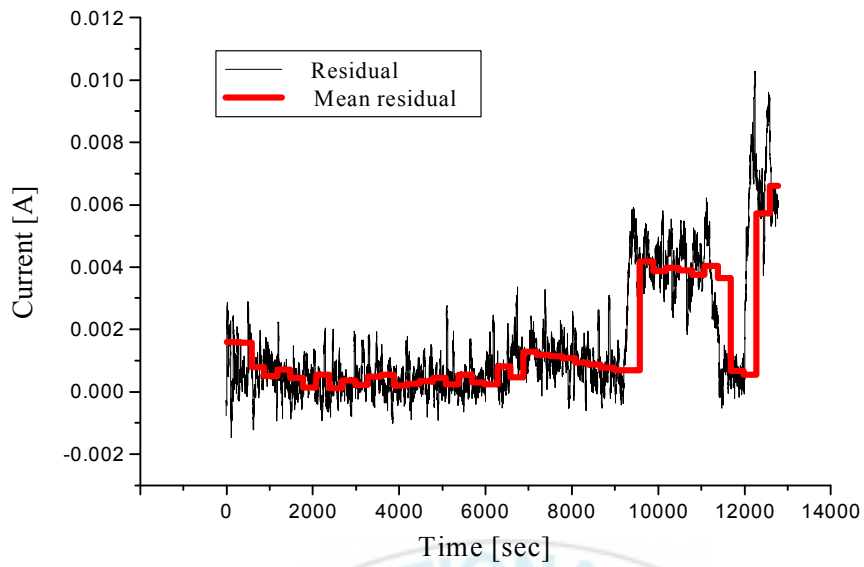
의 변수로서는 적당하지 않음을 알 수 있었다. 하지만 응축온도의 경우 응축기 출구 온도로서 응축전열면적 감소를 잘 반영하였다.

(e)는 응축전열면적 감소에 따른 고장규명 결과이다. NFM(1)을 적용한 고장진단시스템은 응축전열면적 10~20%의 감소는 고장을 검출하지 못했기 때문에 고장이 규명되지 않았다. 하지만 30%이상의 감소에는 정확하게 반응하였고, 그 결과 정확하게 고장이 규명됨을 되었다.

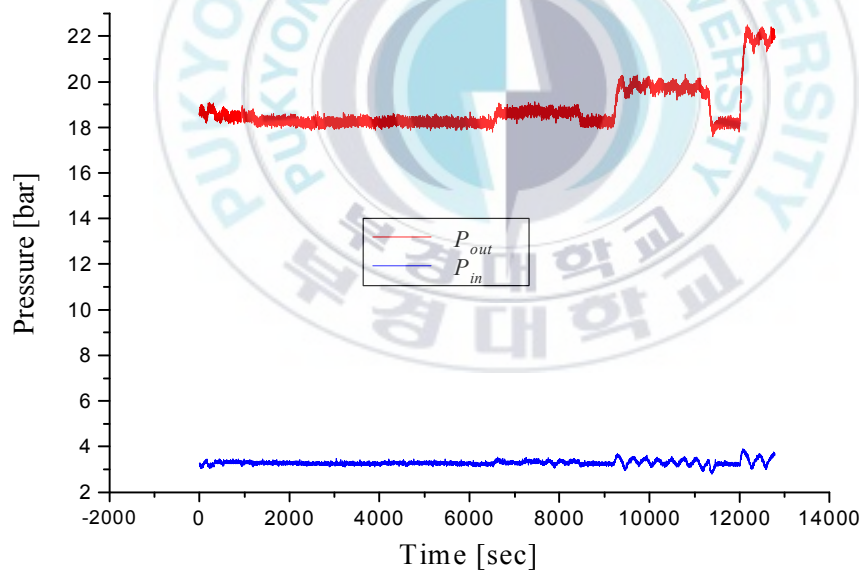
NFM(1)에 의한 FDDS(1)으로 응축기 오염에 대한 실험 결과는 응축전열면적 30% 이상에서는 정확하게 진단하는 신뢰성을 확보하였고, 고장 진단 주기는 300초로 응축기 오염에 대한 FDDS(1)의 속응성도 충분할 것으로 판단되었다.



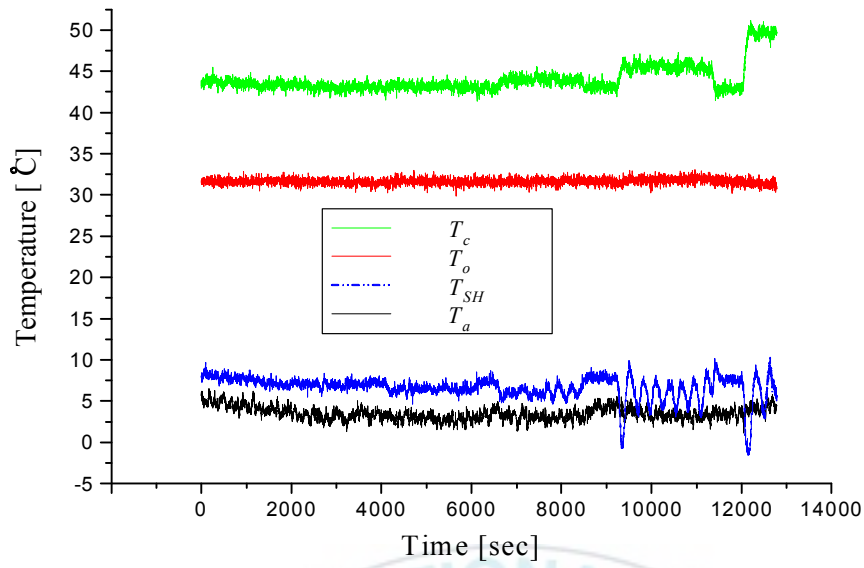
(a) Current information according to change of fault ratio



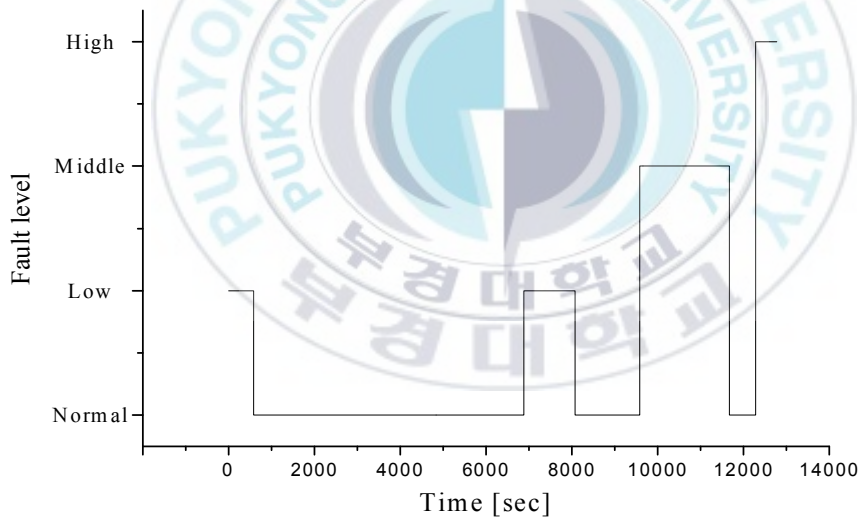
(b) Residual information according to fault ratio



(c) Pressure information according to fault ratio



(d) Temperature information according to fault ratio



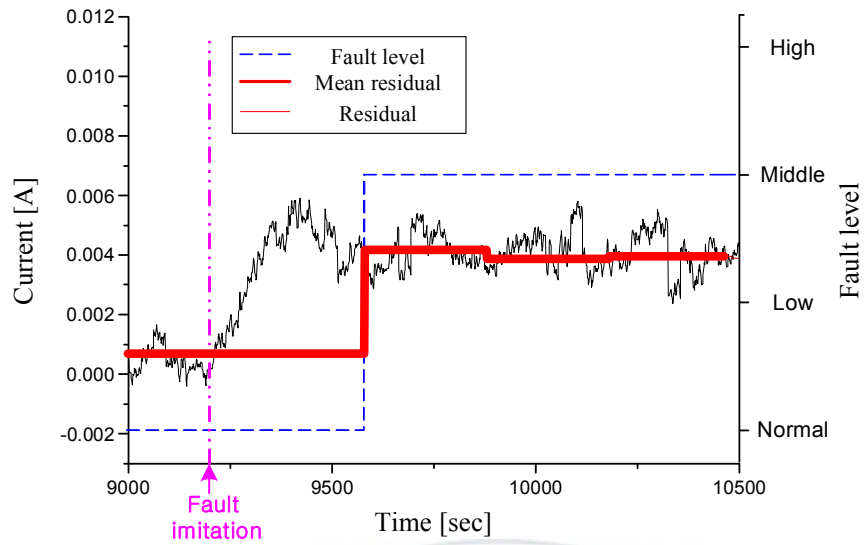
(e) Result of fault identification according to fault ratio

Fig. 12 Result of on-line FDD about condenser fouling in a VSRS

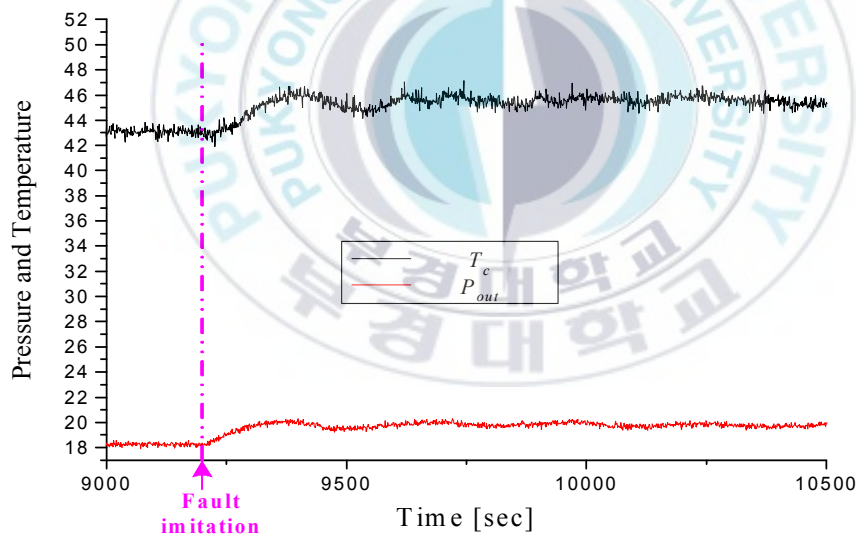
Fig. 13은 응축전열면적 50%감소시킨 직후 전류에 의한 잔차정보와 압력, 온도와의 비교 자료이다.

(a)는 고장 직후 잔차정보와 평균잔차정보 그리고 고장규명결과를 나타낸 것이다. 고장 직후 잔차정보는 급격히 상승한 것을 알 수 있다. 변화 크기는 5mA 정도로 아주 적은 크기이지만, 필터로 인해 안정된 값이고 전류정보의 특성상 이 정도의 크기는 매우 신뢰할 만한 크기이다. 잔차정보의 상승 후 고장진단 주기 300초가 흐른 뒤 평균잔차정보가 상승하여 고장이 규명됨을 알 수 있다.

(b)는 고장 직후 압축기 출구 압력과 응축온도를 나타낸 것이다. 여러가지 압력과 온도 중 두 가지의 경우만 제시한 것은 나머지 압력과 온도는 고장을 진단할만한 특성을 찾기 힘들었기 때문이다. 압축기 출구 압력과 응축온도 역시 고장을 진단할만 정보를 가지고 있다. 이 정보의 조합에 의해 잔차정보가 만들어 졌고, 그에 따라 고장진단이 되었다. 하지만 개개의 정보들은 피드백제어기의 보상효과에 의해 큰 진동의 폭을 보였다. 이는 무고장상태에서 부하의 급변화에 의해 충분히 나타날 수 있는 진동이다. 그러므로 고장 발생 후 진단을 하기 위해서는 진동이 안정화된 후의 시간인 최소 1000초가 필요할 것으로 판단된다. 하지만 전류정보의 결과는 고장 발생 후, 정상상태와 비교되는 크기를 가지고 진동을 하는 것을 알 수 있고, 전류정보를 이용하는 것이 고장진단에 있어서 보다 신뢰성을 가지기 쉬움을 알 수 있었다.



(a) Residual of current right after fault imitation



(b) Pressure and temperature right after fault imitation

Fig. 13 Result right after fault imitation

4.3 가변속냉동시스템 고장진단을 위한 전류정보의 유용성

VSRs에서는 피드백제어기를 사용한다. 피드백제어기는 실내 온도와 과열도를 일정하게 유지하도록 설계되었다. 응축기 전열면적이 감소되면 온도와 압력이 변하게 된다. 이 때 실내 온도와 과열도 역시 변하게 된다. 피드백제어기는 이들 두 값을 일정하게 유지하기 위하여 인버터 주파수와 전자팽창밸브 개도를 변화시킨다. 이는 실내 온도와 과열도만을 일정하게 유지시키는 것이 아니라 다른 온도와 압력 정보에도 보상효과를 일으킨다. 피드백제어기에 의한 온도와 압력정보의 보상효과를 확인하고, 그 때 NFM(2)에 의한 잔차정보와 온도, 압력정보를 비교하여 고장 진단에서의 잔차정보의 이점을 밝히기 위해 응축기 전열면적 30%를 7500초 시점에 감소시켰다.

Fig. 14 (a)는 응축기 오염을 모사한 전열면적 30% 감소에 따른 인버터 주파수와 전자팽창밸브 개도를 나타낸다. 고장 모사 후 증발기 입구 온도가 급격히 상승하고, 과열도가 급격히 떨어졌다. 그 결과 피드백제어기에 의해 전자팽창밸브의 개도는 줄어들게 되었다. 인버터 주파수 역시 변화하는 실내 온도를 설정 온도로 유지하기 위해 변화하는 것을 볼 수 있었다.

Fig. 14 (b)는 응축기 오염에 따른 응축 온도를 나타낸다. 응축 온도만을 제시한 이유는 실내 온도와 과열도는 피드백제어기의 제어대상으로서

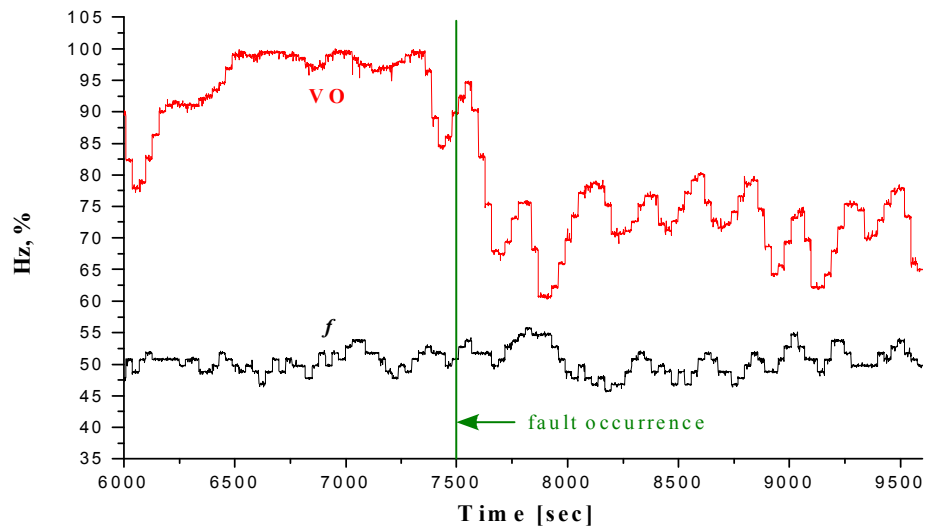
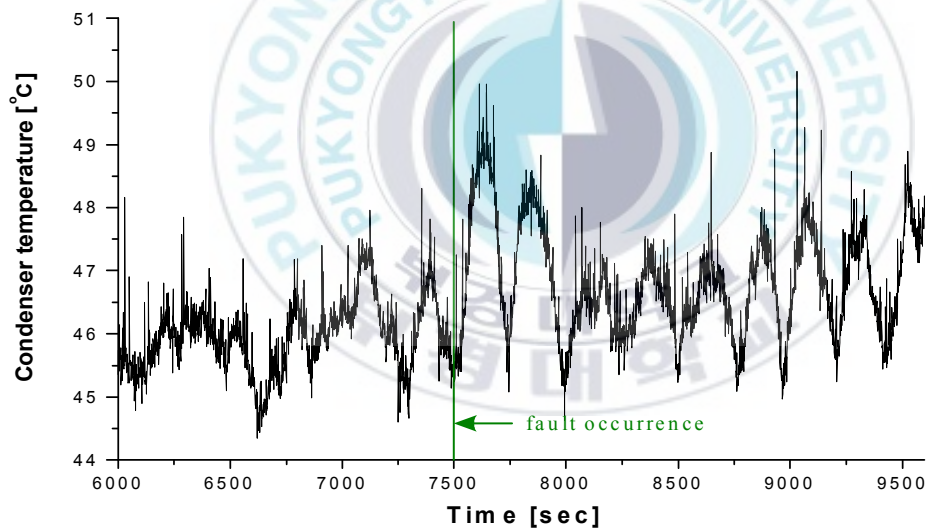
일정하게 유지되어 고장 진단을 위한 정보로는 의미가 없기 때문이다. 고장 모사후 응축기는 외부와 열교환을 충분히 하지 못해 응축 온도가 상승하였다. 그로 인한 VSRS의 실내온도와 과열도는 변하게 되고, 이에 피드백제어기는 온도와 압력에 대해 보상효과를 일으켜 응축온도가 큰 폭으로 진동하였다. 그리고 고장 모사 후 500초가 지난 8000초부터는 고장 전의 응축온도와 비교하여 큰 차이가 없었다. 또한 온도 정보의 경우 주변 환경에 따라 민감하게 변하므로 Fig. 14 (b)의 결과로써 VSRS의 고장 진단을 위한 온도정보는 신뢰성을 확보하기 힘들다는 것을 알 수 있다.

Fig. 14 (c)는 응축기 오염에 따른 압축기 토출압력을 나타낸 것이다. 압축기 흡입압력의 경우 고장 전과 후의 변화가 미미하였다. 하지만 압축기 토출압력은 응축기 오염에 대하여 민감하게 반응하였다. 고장 모사 후 압축기 토출압력은 상승하였으나 응축 온도와 마찬가지로 8000초 이후에는 일정수준으로 유지되는 것을 볼 수 있었다. 압력정보는 응축온도에 비하여 응축기 오염에 대해 고장 전후의 차이가 다소 있지만 신뢰성을 확보할만한 차이는 아니다.

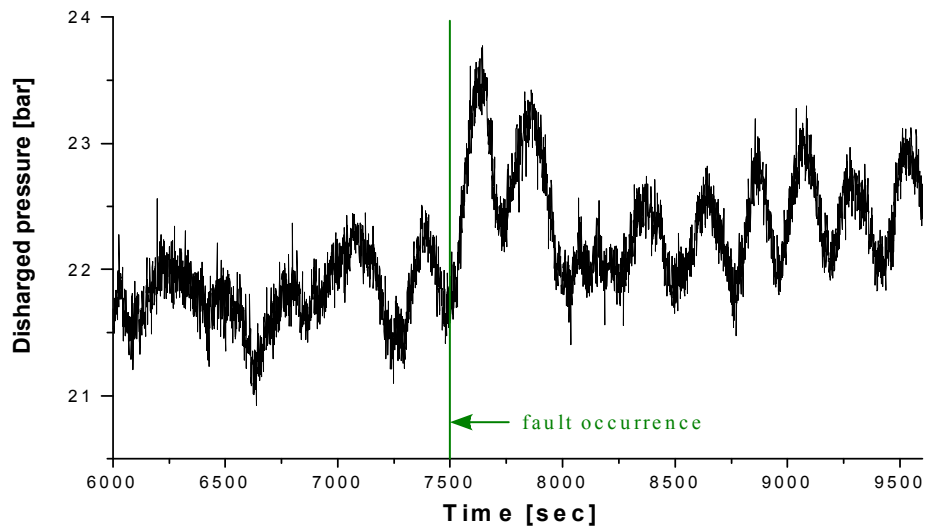
Fig. 14 (d)는 응축기 오염에 따른 잔차정보를 나타낸 것이다. 잔차정보는 압축기의 부하의 크기에 따라 변화하게 된다. 압축기의 부하는 곧 냉매의 상태와 직결되므로 잔차정보는 고장 모사 후 상승하고, 온도와 압력과 마찬가지로 보상효과에 의해 일정한 값으로 수렴한다. 하지만 잔차

정보는 온도와 압력정보와는 다르게 고장 모사 직후부터 고장 모사 전과 비교할 수 있을 정도의 차이를 보였다. 대략 고장 전후의 평균값은 $0.2\sim 0.3[A]$ 차이가 났다. 이 값은 NFM(2)의 f , VO 와 I_d 에 의한 것이므로 오직 전류정보에 기인한 것이다. 따라서 전류정보의 특성을 고려해 볼 때 고장 전후의 차이는 충분히 신뢰성을 가질 수 있다.

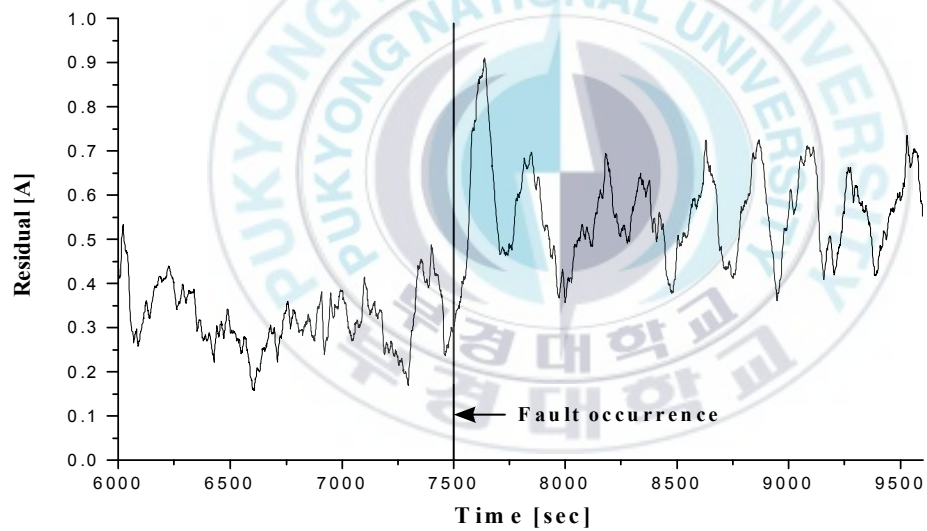


(a) f , VO 

(b) Condenser temperature



(c) Discharged pressure



(d) Residual

Fig. 14 The results according to condenser fouling

4.4 실험 결과(2)

NFM(2)에 의한 FDDS(2)를 이용하여 VSRs의 응축전열면적 감소와 증발기 팬 고장에 따른 고장진단을 수행하였다.

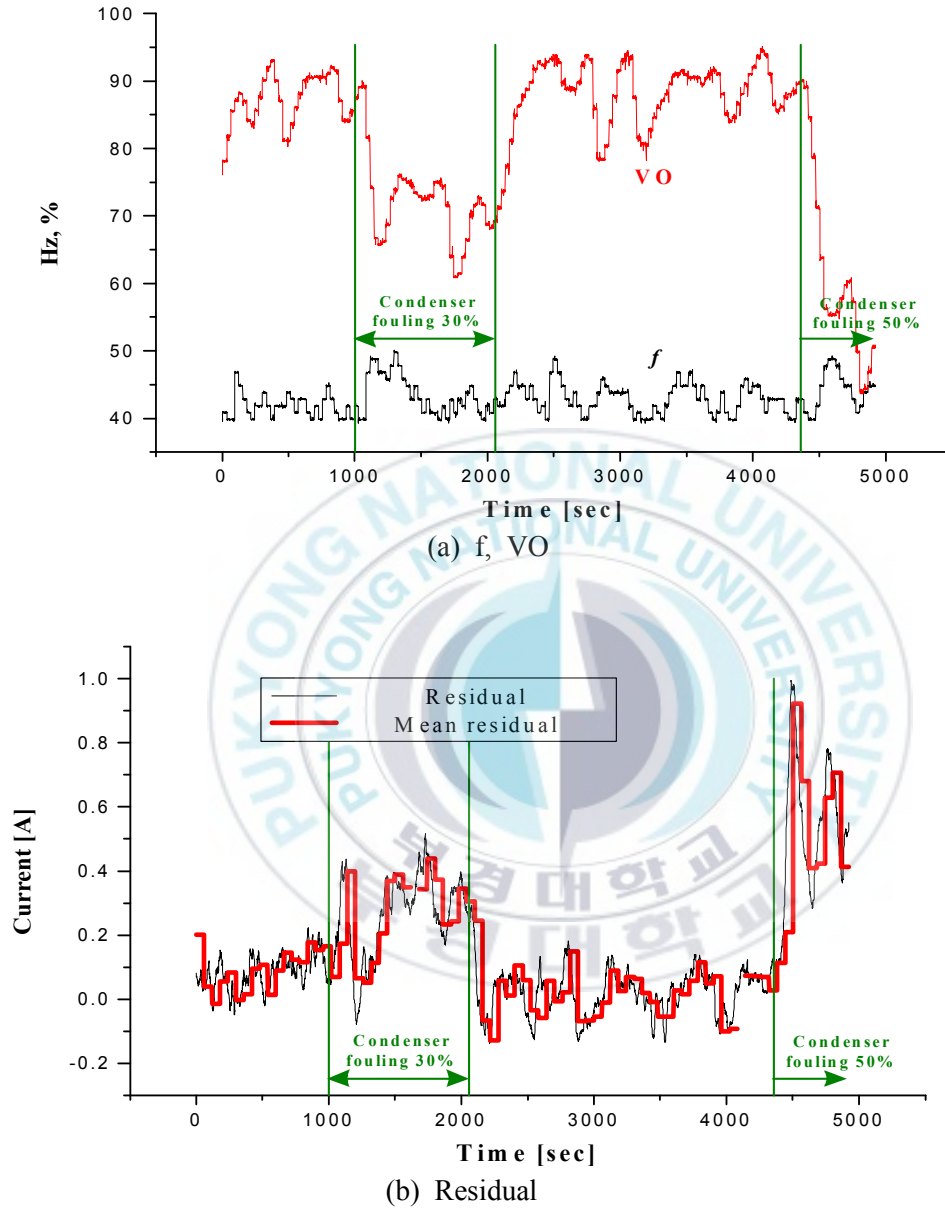
Fig. 15는 전류정보만을 이용한 VSRs의 응축기 오염에 대한 고장 진단 결과이다. 1000~2050초, 4400~4900초에 응축기 전열면적 30%, 50%를 각각 감소시켰다.

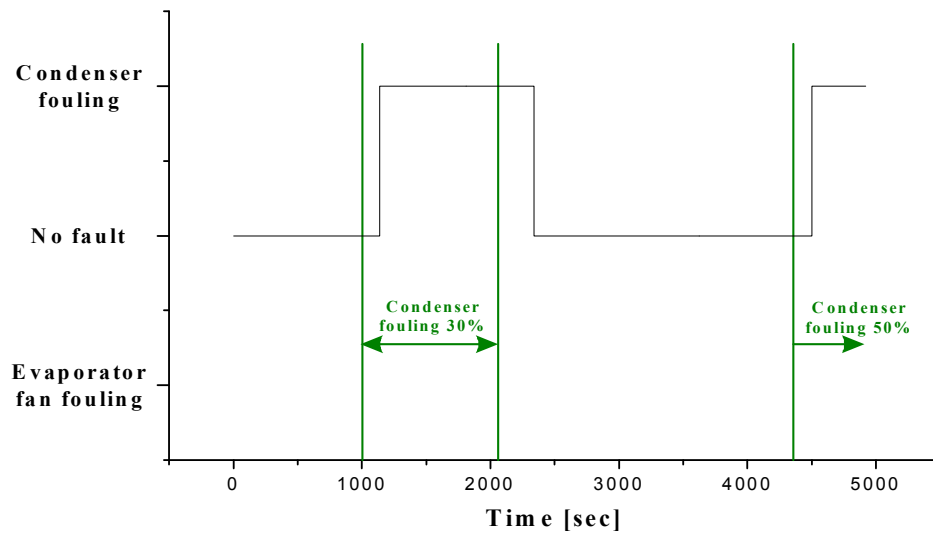
Fig. 15 (a)는 인버터 주파수와 전자팽창밸브 개도의 변화를 나타낸다. 고장의 정도가 클수록 전자팽창밸브 개도가 급격히 감소하고 감소량이 커짐을 알 수 있었다.

Fig. 15 (b)는 응축기 오염에 따른 잔차정보와 평균잔차정보를 나타낸다. 고장의 정도가 커짐에 따라 잔차정보의 크기가 커짐을 알 수 있었고, 그에 따라 고장 진단에 이용될 전류 평균잔차정보 역시 커짐을 알 수 있었다. 평균잔차정보는 충분한 신뢰성을 가짐을 알 수 있었고, 이 정보는 고장진단에 이용되었다.

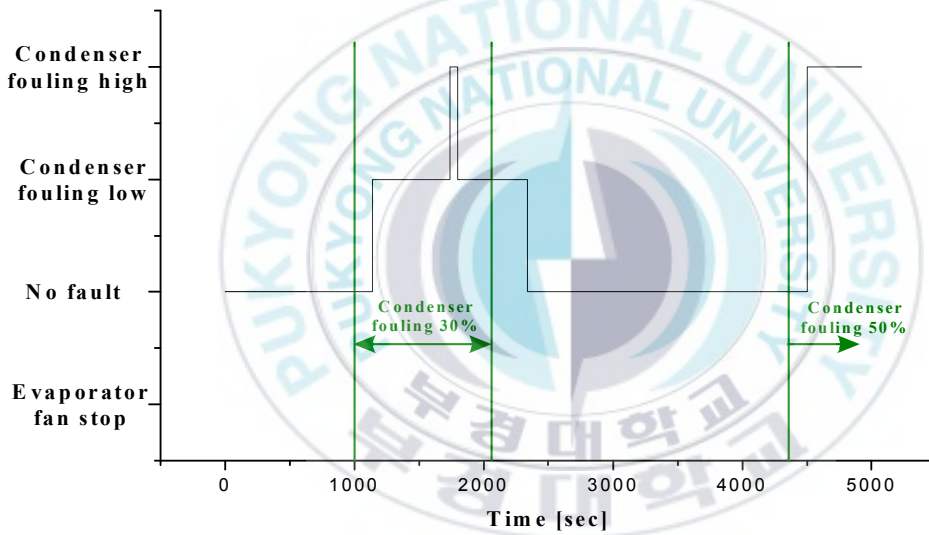
Fig. 15 (c)와 (d)는 고장식별과 고장규명 결과를 각각 나타낸다. 고장 식별 결과 고장 모사 후 정확하게 고장을 식별해 내었다. 그리고 고장 규명 결과 역시 고장 모사 후 정확하게 고장의 정도를 판별해 내었다. 하지만 1740초에 한 번 오진을 한 것을 볼 수 있으나, 이는 고장 진단의 안정화를 위한 필터설계로 쉽게 해결될 수 있을 것으로 판단된다.

Fig. 15의 결과로서 VSRS의 응축기 오염에 대하여 FDDS(2)는 충분한 신뢰성과 속응성을 가지는 것을 알 수 있었다.





(c) Result of fault isolation



(d) Result of fault identification

Fig. 15 On-line FDD result of VSRS corresponding to condenser fouling

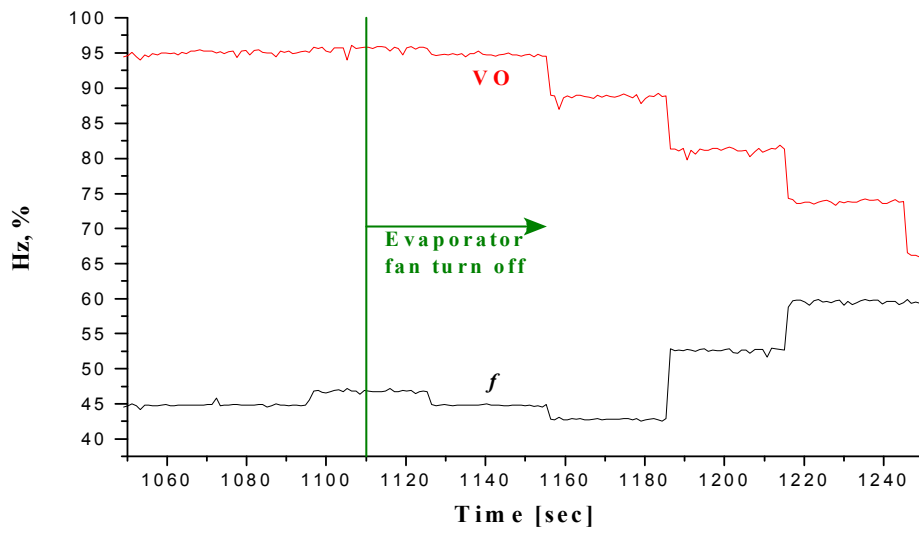
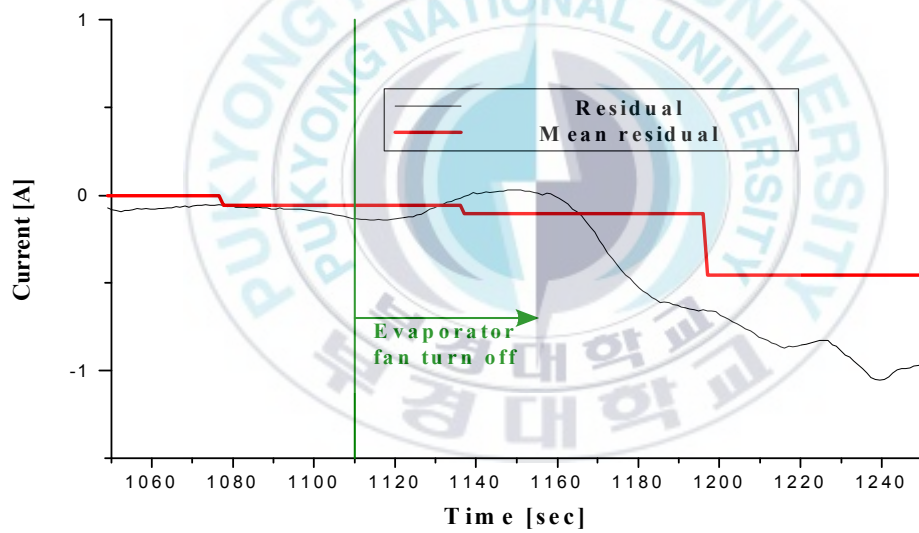
Fig. 16은 FDDS(2)를 이용한 VSRs의 증발기 팬 고장에 대한 고장 진단 결과이다. 1150초에 증발기 팬 고장을 모사했다.

Fig. 16 (a)는 인버터 주파수와 전자팽창밸브 개도의 변화를 나타낸다. 증발기 팬 고장 이후 증발기 출구 온도 감소에 의한 과열도 감소로 전자팽창밸브 개도가 줄어들고 실내 온도의 상승에 따라 인버터 주파수가 상승함을 알 수 있었다.

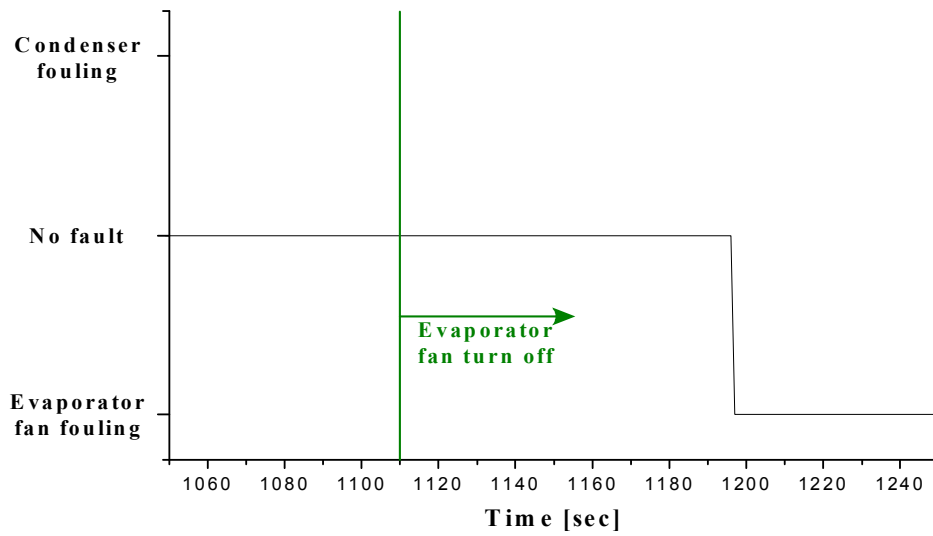
Fig. 16 (b)는 잔차정보와 평균잔차정보를 나타낸다. 증발기 팬 고장의 경우 전류 잔차는 급격한 하락을 보였다. 이는 냉매가 증발기에서 열교환이 되지 않아 압축기의 부하를 줄여줬기 때문이다. 증발기 팬 고장과 같은 경우 압축기의 액압축의 문제가 발생할 수 있으므로 빠른 조치가 필요하다. 잔차정보는 60초마다 고장을 진단하므로 온도와 압력 정보보다 빠른 정보를 제공해 시스템 손상을 최소화할 수 있다.

Fig. 16 (c)와 (d)는 고장식별과 고장규명 결과를 나타낸다. 증발기 팬 고장의 경우 냉동시스템에 큰 무리를 주어 계속적으로 실험을 진행할 수 없었다. 그러나 짧은 시간에도 전류 정보는 고장의 위치와 발생 여부를 정확하게 파악할 수 있음을 알 수 있었다.

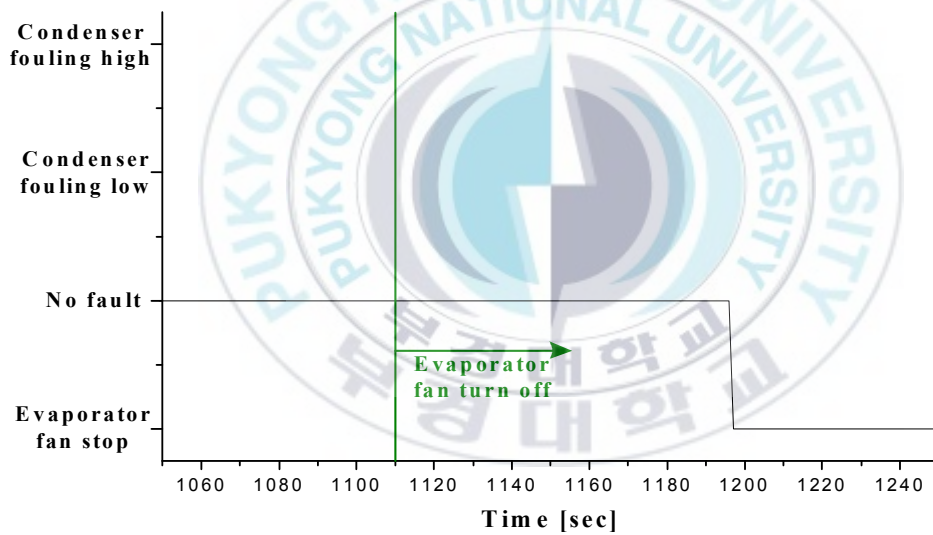
Fig. 16의 결과로서 VSRs의 증발기 팬 고장에 대해서도 전류정보를 이용하여 고장 진단이 가능함을 알 수 있었다.

(a) f , VO 

(b) Residual



(c) Result of fault isolation



(d) Result of fault identification

Fig. 16 On-line FDD result of VSRS corresponding to evaporator fan fouling

4.5 p-h선도를 이용한 열교환기 고장시의 전류 변화 해석

제 4장에서 열교환기 고장에 대하여 응축기 전열면적 감소와 증발기 팬고장에 대하여 전류정보를 이용한 고장진단 결과를 보였다. 압축기의 전류정보가 변하는 원리는 어릴 때 가지고 놀던 DC모터가 달린 장난감 자동차를 생각해 보면 쉽다. 자동차 바퀴를 잡은 채 전원을 On시키면 모터가 타는 것을 본 적이 있을 것이다. 이것은 모터의 부하가 엄청나게 걸려 모터가 돌지 않는 상황이다. 그 때 모터의 역기전력은 0와 가까워지고, 전류는 ∞ 에 가까워지게 되어 모터가 타게 된다. 반대로 모터의 부하가 없다면 역기전력이 상승하여 전류는 낮아지게 된다. 이 원리가 냉동시스템의 압축기에도 적용된다. 즉, 압축기의 일량에 의해 전류 실효값의 증가 혹은 감소가 결정된다.

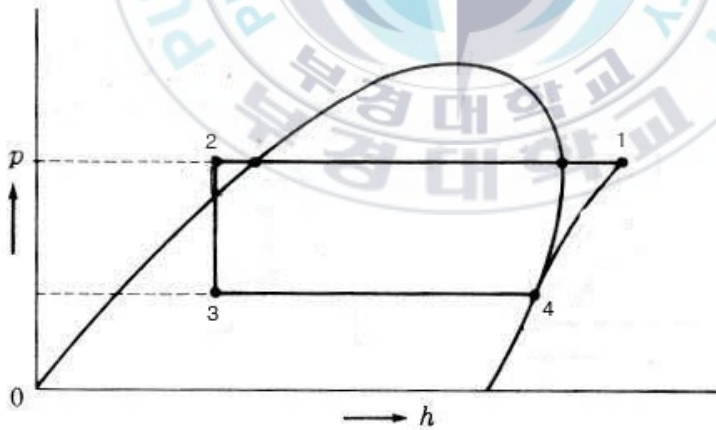


Fig. 17 p-h diagram of basic refrigeration system

Fig. X는 기본 냉동사이클의 이상적인 모리엘선도를 나타낸다. 여기서 압축기의 일량은 $h_1 - h_4$ 이며 COP는 압축기 일량을 냉동능력으로 나누어 준 $\frac{h_4 - h_3}{h_1 - h_4}$ 이다.

Fig. 18와 Fig. 19는 응축기 전열면적 감소와 증발기 팬 고장에 따른 모리엘선도의 개념도이다.

Fig. 18은 응축기 전열면적 감소에 따른 p-h선도의 변화이다. 응축기 전열면적이 감소하게 되면 응축 불량이 되어 점 2는 점 2'로 이동하게 된다. 이 때 증발기 입구온도가 상승하게 되어 과열도가 줄어들게 되고, 피드백제어기에 의해 전자팽창밸브 개도는 줄어들게 된다. 따라서 냉동 시스템에 흐르는 냉매량이 줄어들게 되어 냉동능력이 줄어들게 된다. 냉동능력이 줄어들게 되면 실내온도가 상승하게 되므로 피드백제어기에 의해서 압축기 회전수는 증가하게 된다. 결론적으로 p-h선도는 1'→2'→3'→4와 같이 그려지게 된다. 압축기 일량이 $h_1' - h_4$ 가 되게 되어 증가하게 된다. 그 결과 압축기 모터의 역기전력이 줄어들게 되고, 전류는 증가하게 된다. 그리고 COP는 압축기 일량은 늘어나고, 냉동능력은 줄어들게 되어 줄어들게 된다.

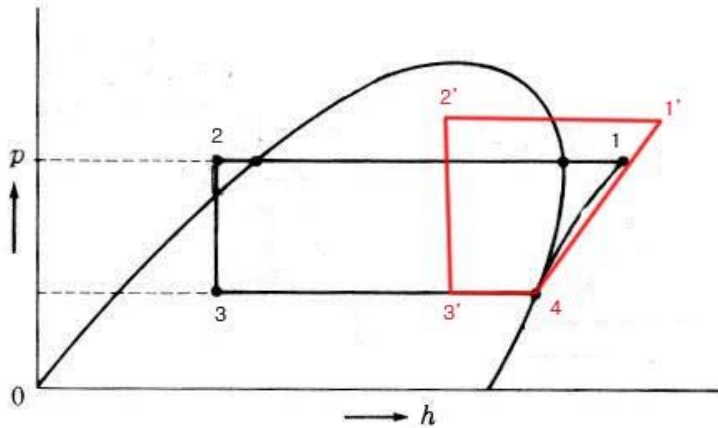


Fig. 18 Change of p-h diagram according to condenser fouling

Fig. 19는 증발기 팬 고장에 따른 p-h선도의 변화이다. 증발기 팬이 고장나서 정지하게 되면 실내의 부하를 냉매가 흡수하지 못하게 된다. 따라서 증발기 출구의 점 4는 점 4'로 이동하게 된다. 이 때 증발기 출구 온도가 감소하게 되어 과열도는 감소하게 된다. 따라서 피드백제어기에 의해 전자팽창밸브의 개도는 줄어들게 되고, 냉매량은 줄어들게 된다. 그리고 증발기에서 열교환이 되지 않으므로 실내온도는 상승하게 된다. 그 결과 피드백제어기에 의해 압축기 회전수는 상승하게 된다. 이에 따라 증발기 팬 고장이 발생하면 냉동사이클은 $1' \rightarrow 2' \rightarrow 3' \rightarrow 4'$ 와 같이 그려지게 된다. 압축기 일량은 등엔트로피 과정이기 때문에 냉매의 액과 기체 혼합 영역에서는 압축기의 일량은 현저히 줄어들게 된다. 따라서 압축기 일량이 줄어들었기 때문에 압축기 모터의 역기전력은 증가하게 되고, 전류는 줄어들게 된다.

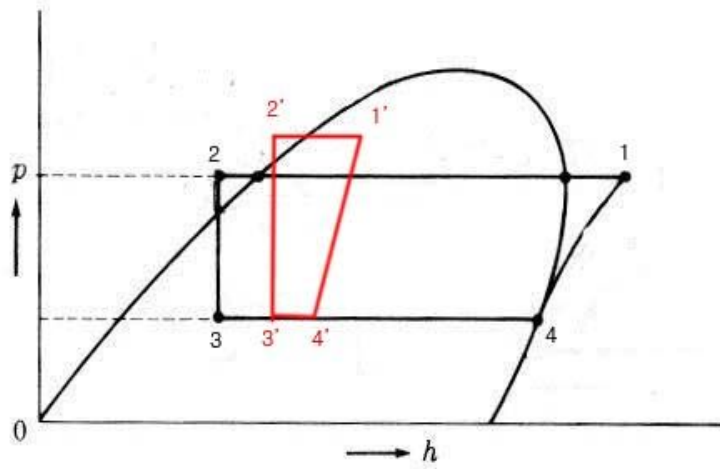


Fig. 19 Change of p-h diagram according to evaporator fan fouling



제 5 장 결 론

본 논문에서는 가변속냉동시스템(VSRS)의 열교환기 고장에 대해 전류 정보를 이용한 실시간 고장검출과 고장진단 알고리즘을 구성하고, 고장 진단시스템(FDDS)을 구축하였으며, 실험을 통해 제안된 FDDS의 효용성을 검증하였다.

고장검출을 위해 NFM(1)과 NFM(2)의 두 모델을 제안하고 이 두 모델에 대한 각각의 FDDS를 구성하여 열교환기 고장에 대한 실험을 한 결과, FDDS(1)은 응축기 전열면적 감소 30%이상, FDDS(2)는 응축기 전열면적 감소 30%이상과 증발기 팬 고장에 대하여 정확하고 빠른 FDD 결과를 확인하였다.

FDDS는 고장에 대한 신뢰성과 속응성 확보가 대단히 중요하다. FDDS의 성능인 신뢰성과 속응성은 FDD를 위한 모델의 변수 선택에 따라 결정되므로 이 두 성능은 모델과 상호 긴밀한 연관성을 가진다. NFM(1)은 인버터 주파수, 온도, 압력정보를 사용하였고, NFM(2)는 온도와 압력정보를 배제하고 인버터 주파수와 전자팽창밸브개도 정보를 사용하였다. 각 NFM에 의한 FDDS의 실험 결과, FDDS(1)과 FDDS(2)의 신뢰성은 열교환기 고장 모사 결과로 확보됨을 확인하였으나, 신뢰성 확보를 위해 필요한 고장진단 주기를 검토한 결과 FDDS(2)의 경우가 FDDS(1)에 비해 주기를 5배 짧게 설정할 수 있는 것으로 나타나 결과적으로 속응성이 더

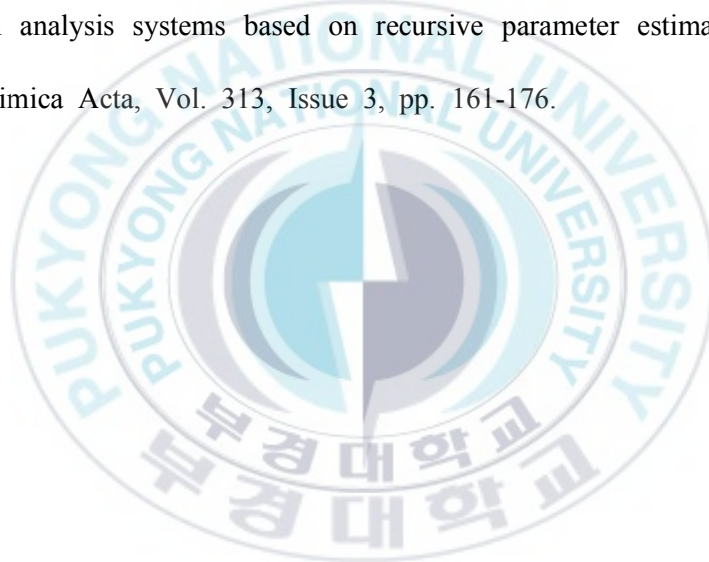
뛰어남을 알 수 있었다. 이는 NFM(2)에서 큰 시정수와 피드백제어기의 보상 효과를 지닌 온도 및 압력정보를 모델에서 배제시켰고, 고장 정보 추출을 위한 측정 센서의 수도 8개에서 1개로 줄여 각 정보의 잡음 축소를 위한 노이즈 필터링 처리 시간이 최소화될 수 있었기 때문으로 해석된다. 이리하여 NFM(2)를 이용한 FDDS(2)는 VSRS의 열교환기 고장 상태를 압축기의 부하에 따른 전류정보만을 이용함으로써 더욱 정확하고 빠른 FDD가 가능했다.

본 연구는 VSRS의 기계적인 고장 가운데 열교환기인 응축기와 증발기 두 기기에 국한해서 고장진단을 하였기 때문에 전류정보만으로도 가능하였다. 그러나 고장진단 대상 기기의 수가 많아진다면 고장진단부의 고장 식별과 고장규명을 위해서 전류뿐만 아니라 다른 정보도 적극적으로 이용하는 방안을 고려해야 될 것으로 생각된다. 그 이유는 전류가 가지는 정보만으로는 다양한 기기의 고장진단에 한계가 있을 것으로 생각되기 때문이다. 그리고 본 연구에서는 고장 모사 시, 각 기기의 단독 고장만을 취급하였지만 여러 기기가 다양한 고장 정도로 고장을 일으키고 있는 복합적인 고장에 대한 연구가 향후 필요할 것으로 생각된다. 또한 제안된 VSRS의 기계적인 고장진단시스템을 더욱 발전시켜 압축기의 전기적인 고장진단까지를 포함하는 종합적인 VSRS의 고장진단 시스템 구축에 대한 연구도 필요할 것으로 생각된다.

참 고 문 헌

1. D. G. Lee, D. G. Koh and S. K. Jeong, 2006, On-line FDD of Condenser Fouling in a Variable Speed Refrigeration System Based on Current Information, Proceed of KSPSE, pp. 150-155
2. J. S. Yoon and S. K. Jeong, 2006, On-line FDD for a Variable Speed Refrigeration System Based on Current Information Analysis of a Compressor, Proceed of SAREK, pp. 434-439
3. T. G. Paek, J. S. Ryu and K. S. Lee, 2002, Real-time FDI Schematic for AC Motor Control Systems, Proceed of IEEE, pp. 77-81.
4. Li Hua, D. W. Lee, S. K. Jeong and J. I. Yoon, 2006, Fuzzy Control with Feedforward Compensator of Superheat in a Variable Speed Refrigeration System, Journal of the Korean Society of Marine Engineering, Vol. 31, No. 4, pp. 369-379.
5. D. Y. Han and H. H. Lee, 2002, Partial Fault Detection of an Air-conditioning System by using the Model-based Method with Data Preprocessing, Proceed of SAREK, pp. 295-300.
6. M. S. Kim and M. S. Kim, 2005, Studies on the Performance Variation of a Variable Speed Vapor Compression System under Fault and Its Detection and Diagnosis, Proceed of SAREK, Vol. 17, No. 1, pp. 47-55.

-
7. H. I. Shin, S. H. Lee, C. H. Woo, S. J. Kang, H. B. Kim and K. B. Woo, 1997, A Study on Fault Detection and Diagnosis of Induction Motor by Clustering and Fuzzy Fault Tree, Proceed of IEEK, pp. 173-176.
 8. Jafar Zarei and Javad Poshtan, 2006, Bearing fault detection using wavelet packet transform of induction motor stator current, Tribology International, Vol. 40, No. 5, May 2007, pp. 763-769.
 9. Xiaon Wu, K. H. Bellgardt, 1995, "On-line fault detection of flow-injection analysis systems based on recursive parameter estimation", Analytica Chimica Acta, Vol. 313, Issue 3, pp. 161-176.



학술대회 발표 논문 List

- 1) 이동규, 윤종수, 이상호, 금종수, 정석권, 2006, 터널의 환기제어를 위한 퍼지 제어기 설계, Fuzzy controller design for ventilation control of tunnel, Proceed of KSPSE, pp. 292~299
- 2) 이동규, 정석권, 장대 터널의 효율적인 환기를 위한 퍼지제어기 설계, 2006, Fuzzy Controller Design for Efficient Ventilation in a Long Size Tunnel, Proceed of SAREK, pp. 105
- 3) Jong-Su Yoon, Dong-Gyu Lee, Seok-Kwon Jeong, 2006, On-line Fault Diagnosis of a Refrigeration system Using Analysis of the harmonics on the Current Information, Proceed of ICCHT
- 4) 이동규, 고대권, 정석권, 2007, 전류정보를 이용한 가변속냉동시스템의 응축기 오염 On-line 고장진단, On-line FDD of Condenser Fouling in a Variable Speed Refrigeration System Based on Current Information, Proceed of KSPSE, pp. 150~155
- 5) Dong-Gyu Lee, Dae-Kwon Koh, Seok-Kwon Jeong, 2007, On-line Fault Detection and Diagnosis of Condenser Fouling for a Variable Sped Refrigeration System Based on Current Information, Proceed of ICCHT,

pp. 74~80

- 6) 이동규, 정석권, 전류정보를 이용한 가변속냉동시스템의 열교환기
실시간 고장 진단, On-line Fault Detection and Diagnosis for Heat
Exchanger of Variable Speed Refrigeration System Based on Current
Information, 2007 SAREK, pp. 17



Appendix A 선형회귀분석(Linear regression analysis)

전류정보의 회귀분석은 식 (1), (2)의 선형회귀분석법을 사용하였다.

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_n x_{ni} + \epsilon_i \quad (1)$$

$$y = X\beta + \epsilon \quad (2)$$

식 (1)을 행렬로 바꾸면 식 (2)와 같이 된다. 여기서 y 는 종속변수이고, X 는 독립변수이다. 오차 제곱의 합을 Q 라 하면 식 (2)는 식 (3)과 같이 된다.

$$\begin{aligned} Q = \sum_{i=1}^n \epsilon^2 &= \epsilon' \epsilon = (y - X\beta)'(y - X\beta) \\ &= y'y - 2\beta' X'y + \beta' X'X\beta \end{aligned} \quad (3)$$

양변을 편미분하면 식 (4)와 같이 된다.

$$\frac{\partial Q}{\partial \beta} = -2y'X + 2\beta'X'X \quad (4)$$

$\frac{\partial Q}{\partial \beta} = 0$ 가 될 때 오차 제곱합이 최소가 되고, 그 때의 β 가 최적의 회귀

계수가 된다. $\beta = b$ 라면,

$$\begin{aligned} X'Xb &= X'y \\ (X'X)^{-1}X'Xb &= (X'X)^{-1}X'y \end{aligned} \quad (5)$$

로 정리되고, 구하고자 하는 회귀계수의 행렬 b 는 식 (6)과 같이 구해진다.

$$b = (X'X)^{-1}X'y \quad (6)$$



Appendix B 퍼지제어기(Fuzzy controller)

B-1. 용량제어

- Rule base(용량제어)

Rule base

	-3	-2	-1	0	1	2	3
-3	-3	-3	-3	-3	0	0	0
-2	-3	-2	-2	-2	0	0	1
-1	-2	-1	-1	-1	0	1	1
0	-2	-1	-1	0	1	1	1
1	-2	-1	0	1	1	1	2
2	-1	0	0	2	2	2	3
3	0	0	0	3	3	3	3

Rule load

B-2. 과열도제어

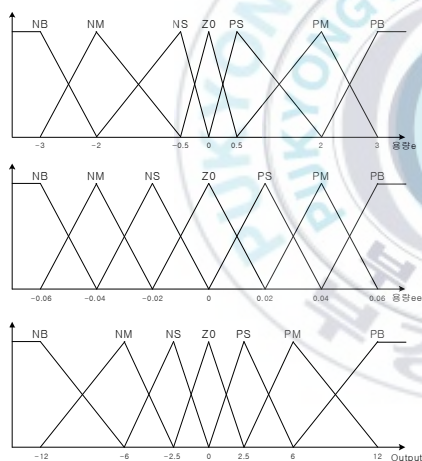
- Rule base(과열도제어)

Rule base

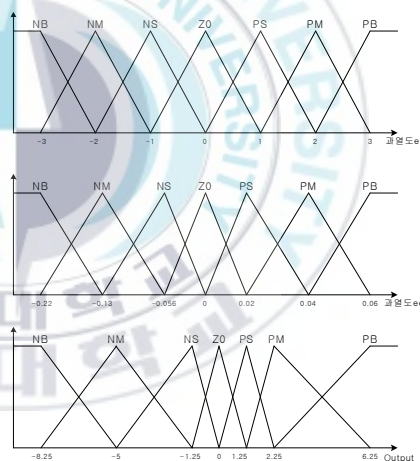
	-3	-2	-1	0	1	2	3
-3	-3	-3	-3	-3	-1	0	0
-2	-3	-3	-3	-2	0	0	0
-1	-3	-2	-2	-1	0	0	1
0	-2	-1	-1	0	1	1	2
1	-1	0	0	1	2	2	3
2	0	0	0	2	3	3	3
3	0	0	1	3	3	3	3

Rule load

- Membership function(용량제어)



- Membership function(과열도제어)



- Look-up table(용량제어)

Look up table													
	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-6	-6	-6	-6	-6	-6	-6	-2	0	0	0	0	0
-5	-6	-5	-5	-5	-5	-5	-5	-2	0	0	0	0	1
-4	-6	-5	-4	-4	-4	-4	-4	-2	0	0	0	1	2
-3	-5	-4	-3	-3	-3	-3	-3	-2	0	0	1	2	2
-2	-4	-3	-2	-2	-2	-2	-2	-1	0	1	2	2	2
-1	-4	-3	-2	-2	-2	-2	-1	0	1	2	2	2	2
0	-4	-3	-2	-2	-2	-1	0	1	2	2	2	2	2
1	-4	-3	-2	-2	-1	0	1	2	2	2	2	2	3
2	-4	-3	-2	-1	0	1	2	2	2	2	2	3	4
3	-3	-2	-1	0	0	2	3	3	3	3	3	4	5
4	-2	-1	0	0	0	2	4	4	4	4	4	5	6
5	-1	0	0	0	0	2	5	5	5	5	5	5	6
6	0	0	0	0	0	2	6	6	6	6	6	6	6

- Look-up table(과열도제어)

Look up table													
	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-6	-6	-6	-6	-6	-6	-6	-3	-2	-1	0	0	0
-5	-6	-6	-6	-6	-6	-5	-5	-3	-1	0	0	0	0
-4	-6	-6	-6	-6	-6	-5	-4	-2	0	0	0	0	0
-3	-6	-5	-5	-5	-5	-4	-3	-2	0	0	0	0	1
-2	-6	-5	-4	-4	-4	-3	-2	-1	0	0	0	1	2
-1	-5	-4	-3	-3	-3	-2	-1	0	1	1	1	2	3
0	-4	-3	-2	-2	-2	-1	0	1	2	2	2	3	4
1	-3	-2	-1	-1	-1	0	1	2	3	3	3	4	5
2	-2	-1	0	0	0	1	2	3	4	4	4	5	6
3	-1	0	0	0	0	2	3	4	5	5	5	5	6
4	0	0	0	0	0	2	4	5	6	6	6	6	6
5	0	0	0	0	1	3	5	5	6	6	6	6	6
6	0	0	0	1	2	3	6	6	6	6	6	6	6

Appendix C 룩업테이블(Look up table) 생성기 소스 (Visual basic)

C-1. main

```
Private Sub Lookupstart_Click()
Lookupcreator.Show
End Sub
```

```
Private Sub Vtstart_Click()
calVt.Show
End Sub
```

C-2. public 변수

```
Public colnumber As Integer
Public colnumber2 As Integer
'Public u As Integer
'Public c As Integer
Public Wi As Single
Public constp As Integer
Public constp1 As Integer
Public constp5 As Integer

Public ruleval(25, 25) As Integer
Public Memval(25, 25) As Single
Public memvalbe(25, 25) As Single

Public Lookup(100, 100) As Integer
Public Lookup1(100, 100) As Single
Public Lookup2(100, 100) As Integer

Public traffic1(1000, 100) As Integer
Public CarVt(1000) As Integer
Public dayrow As Integer

Public mrow(30) As Integer
Public mcol(30) As Integer
Public fuzzyset1(9) As Integer
Public fuzzyset2(9) As Integer
Public fuzzyset3 As Integer
Public membership1(9) As Single
Public membership2(9) As Single
Public membership3(9) As Single
Public RL(20, 20) As Integer

Public Wi2(9, 3) As Single
Public fi(100, 100) As Integer
Public Wi_max(100) As Single
Public fi_max(100) As Integer
Public m2(9, 3) As Integer
Public Quan(25, 8) As Single
```

C-3. Lookupcreator

```
Private Sub rule_Click()

Call RL_Load

Call Go_MFG1

'Test = 1
End Sub

Private Sub RL_Load()

Dim exApp As Excel.Application
Dim wb As Excel.Workbook
Dim ws As Excel.Worksheet

Set exApp = CreateObject("excel.application")
Set wb = exApp.Workbooks.Open("C:\AMCL\RL.xls") =
Set ws = wb.Sheets.Item("sheet1")

Dim s As String

constp = Text100
st = 7 - (constp / 2): en = 7 + (constp / 2)
constp1 = constp + 1

For i = 1 To constp1
For j = 1 To constp1
a = st + i - 1
b = st + j - 1
ruleval(i, j) = ws.Cells(a, b)

Next j
Next i

End Sub

Private Sub Go_MFG1()

MSFlexGrid1.Cols = constp1 + 1
MSFlexGrid1.ColWidth(0) = 25 * 15
For i = 1 To constp1 MSFlexGrid1.Cols - 1
MSFlexGrid1.ColWidth(i) = 25 * 15
Next i
```

```

With MSFlexGrid1
    .Row = 0
    For j = 1 To MSFlexGrid1.Cols - 1
        .Col = j
        .Text = -constp / 2 + j - 1
    Next
End With

constp2 = constp + 2

For i = 1 To constp1
    For j = 0 To constp1
        MSFlexGrid1.Rows = i + 1
        MSFlexGrid1.Row = i
        If j = 0 Then
            MSFlexGrid1.Col = 0
            MSFlexGrid1.Text = -constp / 2 + i - 1
        Else
            MSFlexGrid1.Col = j
            MSFlexGrid1.Text = ruleval(i, j)
        End If
    Next j
Next i

End Sub

Sub MSFlexGridEdit1(Grd As Control, Edt As Control, KeyAscii As Integer)
    Select Case KeyAscii
        Case 0 To 32
            Edt = Grd
            Edt.SelStart = 1000
        Case Else
            Edt = Chr(KeyAscii)
            Edt.SelStart = 1
    End Select

    Edt.Move Grd.Left + Grd.CellLeft, Grd.Top + Grd.CellTop, _
        Grd.CellLeft + Grd.CellWidth, Grd.CellTop + Grd.CellHeight
    Edt.Visible = True
    Edt.SetFocus
End Sub

Sub EditKeyCode1(Grd As Control, Edt As Control, _
    KeyCode As Integer, Shift As Integer)
    Select Case KeyCode
        Case 27
            Edt.Visible = False
            Grd.SetFocus
        Case 13
            Grd.SetFocus
        Case 38
            Grd.SetFocus
            DoEvents
            If Grd.Row > Grd.FixedRows Then
                Grd.Row = Grd.Row - 1
            End If
        Case 40
            Grd.SetFocus
            DoEvents
            If Grd.Row > Grd.Rows Then
                Grd.Row = Grd.Row + 1
            End Select
    End Select
End Sub

Private Sub MSFlexGrid1_GotFocus()
    If Text1.Visible = False Then Exit Sub

    MSFlexGrid1 = Text1
    Text1.Visible = False
End Sub

Sub MSFlexGrid1_KeyPress(KeyAscii As Integer)
    MSFlexGridEdit1 MSFlexGrid1, Text1, KeyAscii
End Sub

Sub MSFlexGrid1_DblClick()
    MSFlexGridEdit1 MSFlexGrid1, Text1, 32
End Sub

Private Sub MSFlexGrid1_LeaveCell()
    If Text1.Visible = False Then Exit Sub

    MSFlexGrid1 = Text1
    Text1.Visible = False
End Sub

Private Sub Text1_KeyDown(KeyCode As Integer, Shift As Integer)
    EditKeyCode1 MSFlexGrid1, Text1, KeyCode, Shift
End Sub

Private Sub Text1_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then KeyAscii = 0
End Sub

'-----
'Membership function
'-----

Private Sub membershipfunction_Click()
    Call MF_Load
    Call Go_MFG2
End Sub

Private Sub MF_Load()
    Dim exApp As Excel.Application
    Dim wb As Excel.Workbook
    Dim ws As Excel.Worksheet

    Set exApp = CreateObject("excel.application")
    Set wb = exApp.Workbooks.Open("C:\AMCL\MF.xls")
    Set ws = wb.Sheets.Item("sheet1")

    constp = Text100
    constp1 = constp + 1
    constp5 = constp * 2 + 1

    For i = 1 To constp5
        For j = 1 To constp1
            Memval(i, j) = ws.Cells(i, j)
        Next j
    Next i
End Sub

```



```

Next j
Next i
End Sub

Private Sub Go_MFG2()
MSFlexGrid2.Cols = constp1 + 1
MSFlexGrid2.ColWidth(0) = 25 * 15
For i = 1 To MSFlexGrid2.Cols - 1
    MSFlexGrid2.ColWidth(i) = 25 * 15
Next i

With MSFlexGrid2
    .Row = 0
    For j = 1 To MSFlexGrid2.Cols - 1
        .Col = j
        .Text = -constp / 2 + j - 1
    Next
End With

constp2 = constp + 2

For i = 1 To constp5
    For j = 0 To constp1
        MSFlexGrid2.Rows = i + 1
        MSFlexGrid2.Row = i
        If j = 0 Then
            MSFlexGrid2.Col = 0
            MSFlexGrid2.Text = -constp + i - 1
        Else
            MSFlexGrid2.Col = j
            MSFlexGrid2.Text = Memval(i, j)
        End If
    Next j
Next i

End Sub

Sub MSFlexGridEdit2(Grd As Control, Edt As Shift
Control, KeyAscii As Integer)
Select Case KeyAscii
    Case 0 To 32
        Edt = Grd
        Edt.SelStart = 1000
    Case Else
        Edt = Chr(KeyAscii)
        Edt.SelStart = 1
    End Select

    Edt.Move Grd.Left + Grd.CellLeft, Grd.Top +
Grd.CellTop, -Grd.CellWidth, Grd.CellHeight
    Edt.Visible = True

    Edt.SetFocus
End Sub

Sub EditKeyCode2(Grd As Control, Edt As Control, Private Sub create1()
-
Integer)
    KeyCode As Integer, Shift As Dim n1 As Integer
    Dim n2 As Integer
    Dim k As Integer
    Dim num As Single
    Dim den As Single
    Dim mem3 As Single
    Dim max1 As Single

    Select Case KeyCode
        Case 27
            Edt.Visible = False
            Grd.SetFocus

            Case 13
                Grd.SetFocus
            Case 38
                Grd.SetFocus
                DoEvents
                If Grd.Row > Grd.FixedRows Then
                    Grd.Row = Grd.Row - 1
                Case 40
                    Grd.SetFocus
                    DoEvents
                    If Grd.Row > Grd.Rows Then
                        Grd.Row = Grd.Row + 1
                    End Select
                End Sub

            Private Sub MSFlexGrid2_GotFocus()
                If Text2.Visible = False Then Exit Sub

                MSFlexGrid2 = Text2
                Text2.Visible = False
            End Sub

            Sub MSFlexGrid2_KeyPress(KeyAscii As Integer)
                MSFlexGridEdit2 MSFlexGrid2, Text2,
                KeyAscii
            End Sub

            Sub MSFlexGrid2_DbClick()
                MSFlexGridEdit2 MSFlexGrid2, Text2, 32
            End Sub

            Private Sub MSFlexGrid2_LeaveCell()
                If Text2.Visible = False Then Exit Sub

                MSFlexGrid2 = Text2
                Text2.Visible = False
            End Sub

            Private Sub Text2_KeyDown(KeyCode As Integer,
                Shift As Integer)
                EditKeyCode2 MSFlexGrid2, Text2, KeyCode,
                Shift
            End Sub

            Private Sub Text2_KeyPress(KeyAscii As Integer)
                If KeyAscii = 13 Then KeyAscii = 0
            End Sub

            '-----
            ' Look up table
            '-----

            Private Sub lookcreate_Click()
                Call create1

                Call Go_MFG5
            End Sub

            Dim n1 As Integer
            Dim n2 As Integer
            Dim k As Integer
            Dim num As Single
            Dim den As Single
            Dim mem3 As Single
            Dim max1 As Single

```



```

Dim max2 As Single
Lookup(e, c) = num / den '무게중심
End If

For e = 1 To constp5
    For c = 1 To constp5
        n1 = 0
        n2 = 0
        num = 0: den = 0
        Next c
    Next e
End Sub

For j = 1 To constp1
    If Memval(e, j) <> 0 Then
        n1 = n1 + 1
        fuzzyset1(n1) = j
        membership1(n1) = Memval(e, j)
    End If
    If Memval(c, j) <> 0 Then
        n2 = n2 + 1
        fuzzyset2(n2) = j
        membership2(n2) = Memval(c, j)
    End If
Next j

x = 0
For R1 = 1 To n1
    For R2 = 1 To n2
        Wi = MIN(membership1(R1), membership2(R2))
        f3 = ruleval(fuzzyset1(R1), fuzzyset2(R2))
        convf3 = f3 + (constp / 2) + 1
        max1 = 0
        max2 = 0
        y = 0
        x = x + 1
        For m = 1 To constp5
            If Memval(m, convf3) <> 0 Then
                y = y + 1
                mem3 = Memval(m, convf3)
                Wi2(x, y) = MIN(Wi, mem3) ' 두번째 min연산
                m2(x, y) = m - constp - 1
                max1 = max1 + Wi2(x, y)
                * m2(x, y)
                max2 = max2 + Wi2(x, y)
            End If
        Next m
        num = num + max1
        den = den + max2
    Next R2
Next R1

If den = 0 Then
    Lookup(e, c) = 0 'num / den
Else
    Lookup(e, c) = num / den '무게중심
End If

'Lookup(e, c) = Lookup1(e, c) * 10

Function MIN(a, b)
    If a >= b Then MIN = b
    If a < b Then MIN = a
End Function

Function MAX(a, b)
    If a >= b Then MAX = a
    If a < b Then MAX = b
End Function

Private Sub Go_MFG5()
    MSFlexGrid5.Cols = constp5 + 1
    MSFlexGrid5.ColWidth(0) = 25 * 15
    For i = 1 To MSFlexGrid5.Cols - 1
        MSFlexGrid5.ColWidth(i) = 25 * 15
    Next i
    With MSFlexGrid5
        .Row = 0
        For j = 1 To MSFlexGrid5.Cols - 1
            .Col = j
            .Text = -constp + j - 1
        Next j
    End With
    constp2 = constp + 2
    For i = 1 To constp5
        For j = 0 To constp5
            MSFlexGrid5.Rows = i + 1
            MSFlexGrid5.Row = i
            If j = 0 Then
                MSFlexGrid5.Col = 0
                MSFlexGrid5.Text = -constp + i - 1
            Else
                MSFlexGrid5.Col = j
                MSFlexGrid5.Text = Lookup(i, j) '
            End If
        Next j
    Next i
End Sub

Private Sub Go_Colsize()
    Dim iCols, data
    On Error Resume Next
    targetFileName = "rulebase.txt"
    Open targetFileName For Input As #1

```

```

While Not EOF(1)
    Line Input #1, data
    data = Split(data, ",")
    iCols = UBound(data) + 1
Wend
Close #1

Text1 = iCols
colnumber = iCols
End Sub

Private Sub Go_ColSize2()
Dim iCols, data

On Error Resume Next
targetFileName = "membership.txt"
Open targetFileName For Input As #1

While Not EOF(1)
    Line Input #1, data
    data = Split(data, ",")
    iCols = UBound(data) + 1
Wend
Close #1

Text2 = iCols
colnumber2 = iCols
End Sub

Private Sub Go_MFG23()
Dim dat2() As Single
ReDim dat2(colnumber2) As Single

n = 1
MSFlexGrid2.Cols = colnumber2 + 1
MSFlexGrid2.ColWidth(0) = 30 * 15
For i = 1 To MSFlexGrid2.Cols - 1
    MSFlexGrid2.ColWidth(i) = 22 * 15
Next i

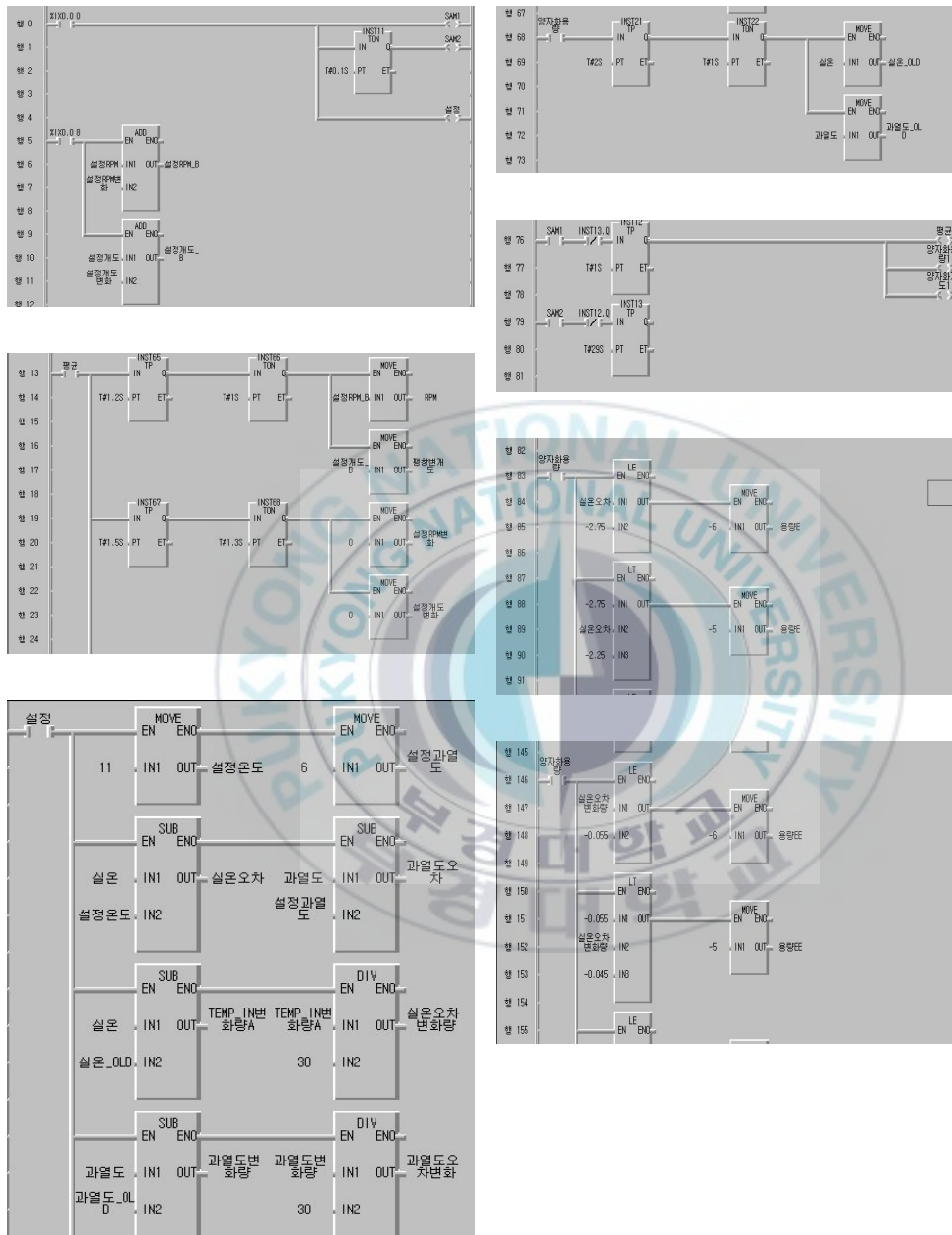
With MSFlexGrid2
    .Row = 0
    For j = 1 To colnumber2 'MSFlexGrid1.Cols - 1
        .Col = j
        .Text = -(colnumber2 / 2 - 0.5) + (j - 1)
    Next j
End With

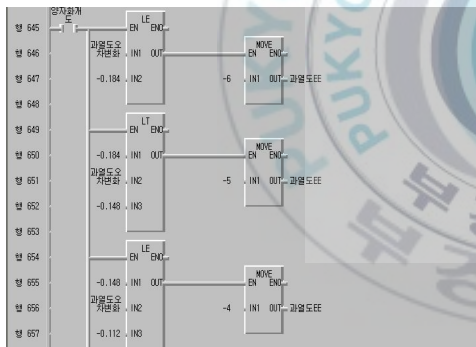
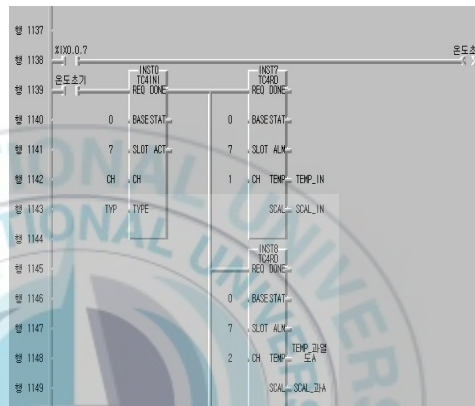
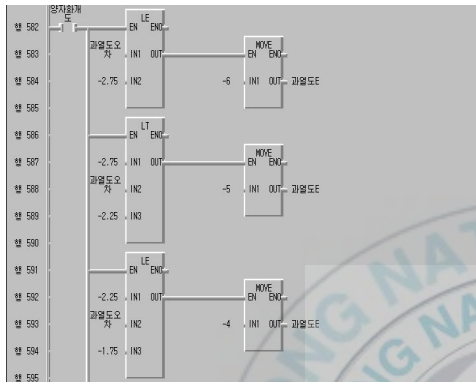
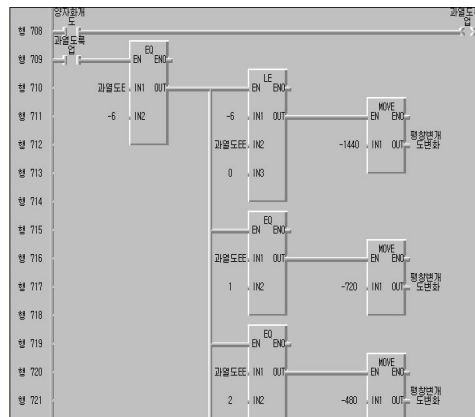
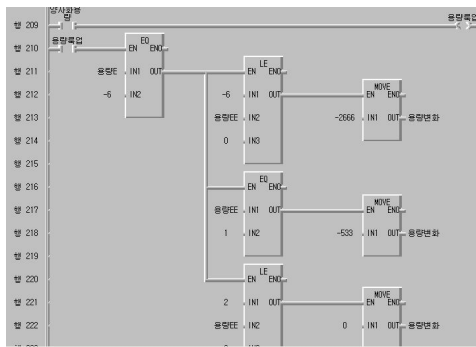
Open "membership.txt" For Input As #1
Do Until EOF(1)
    MSFlexGrid2.Rows = n + 1
    MSFlexGrid2.Row = n

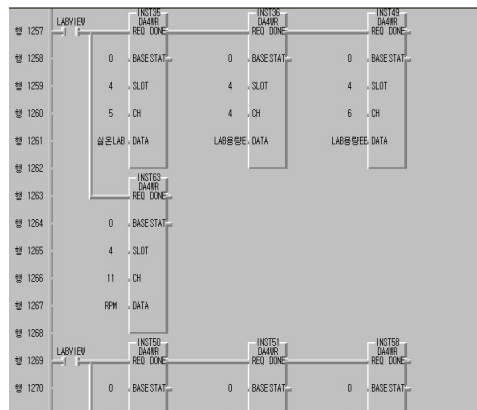
    For i = 0 To MSFlexGrid2.Cols - 2
        Input #1, dat2(i)
        MSFlexGrid2.Col = 0
        MSFlexGrid2.Text = -colnumber2 + n
        MSFlexGrid2.Col = i + 1
        MSFlexGrid2.Text = dat2(i)
        Memval(n, i + 1) = dat2(i)
    Next i
    n = n + 1
    Loop
Close #1
End Sub

```

Appendix D 펄지제어기 PLC 소스(Ladder diagram)

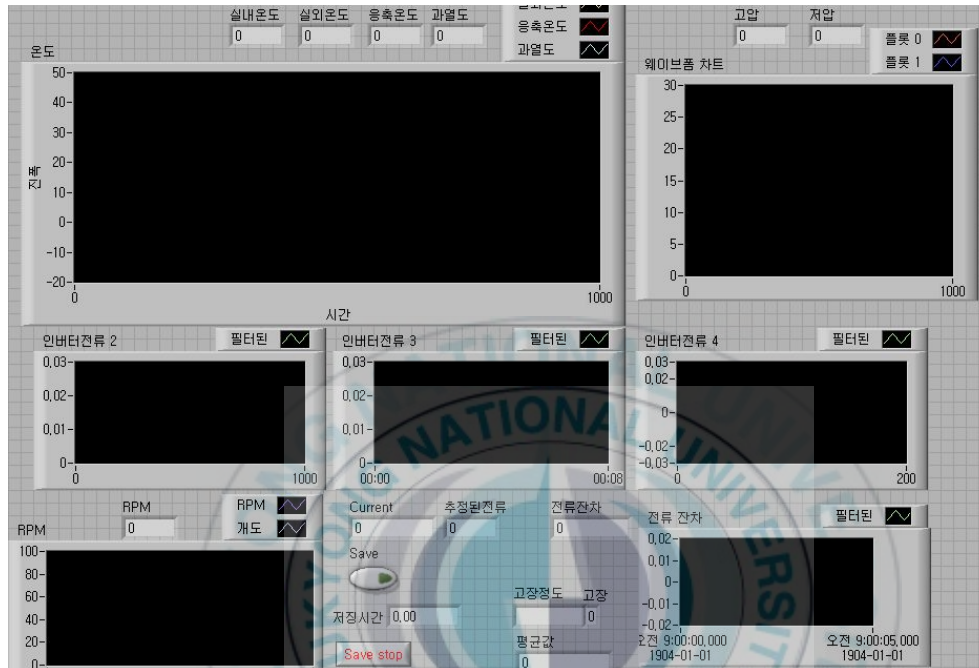


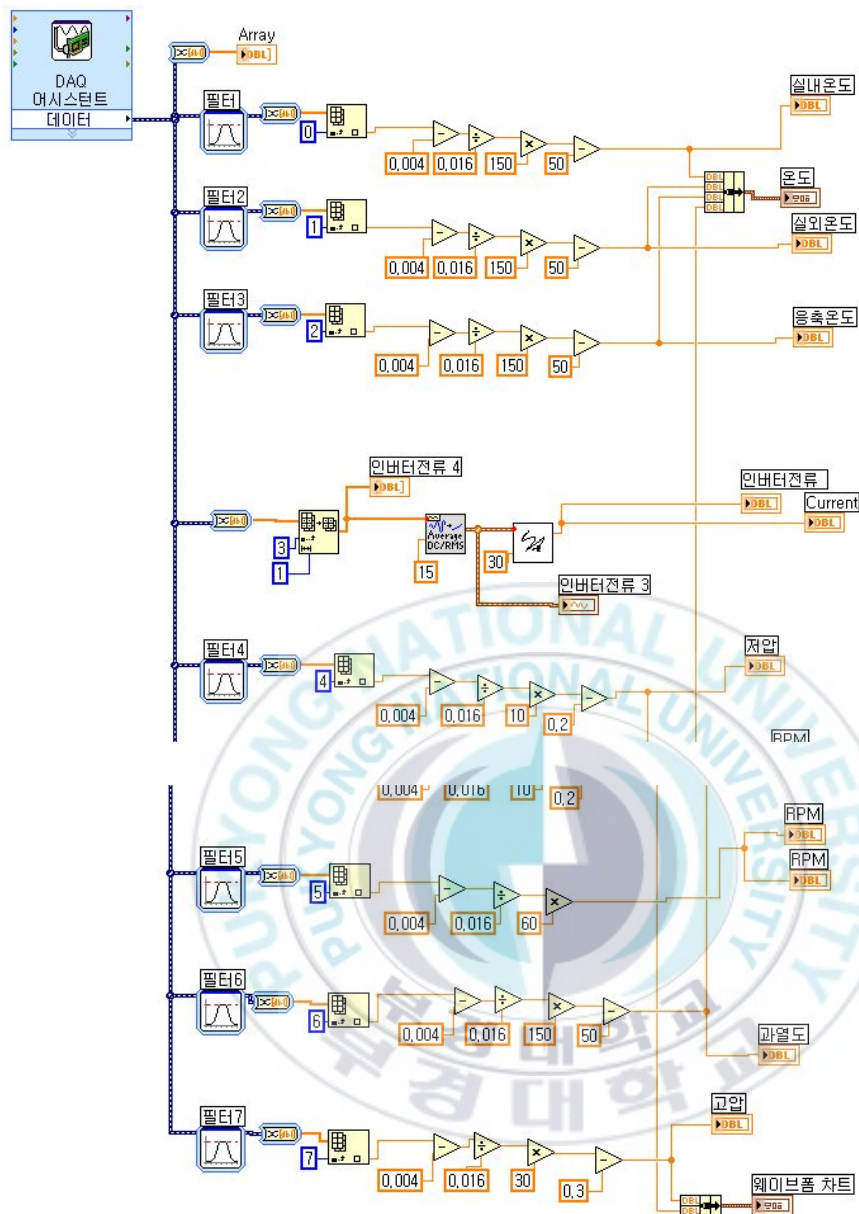


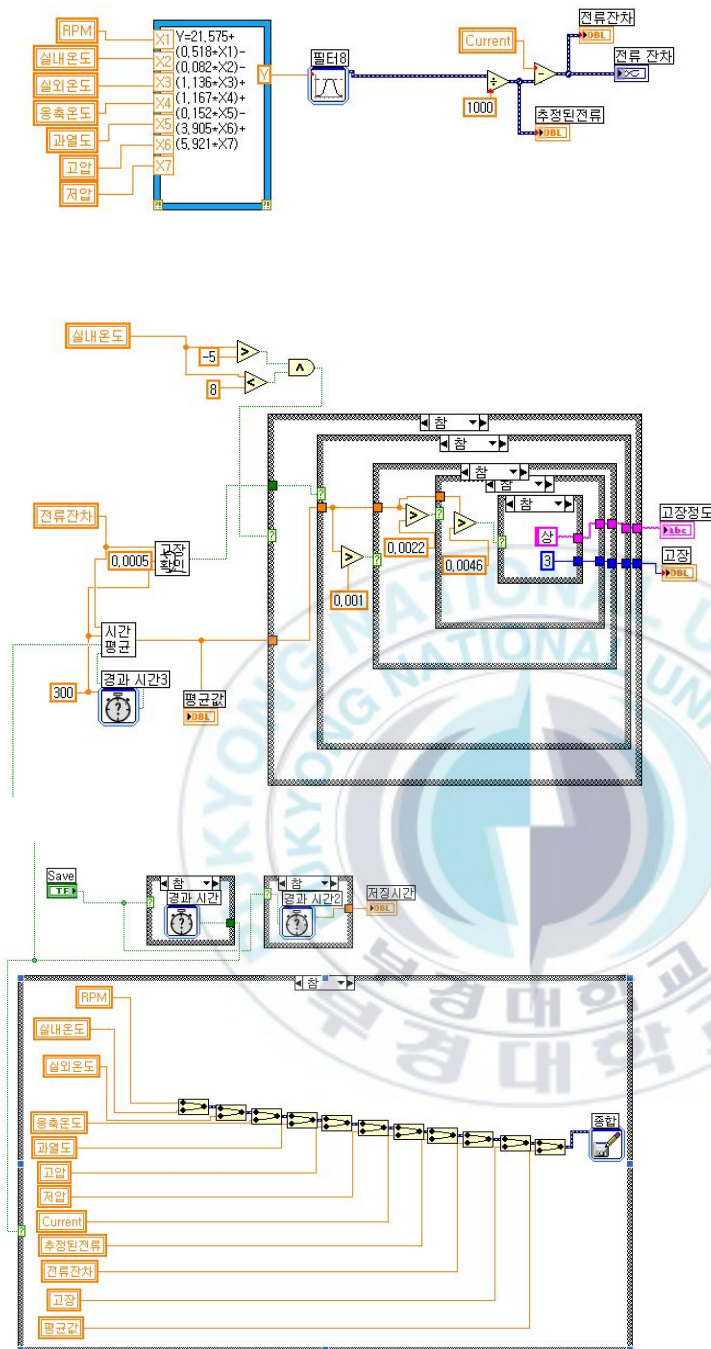


Appendix E 고장진단 소스(Labview)

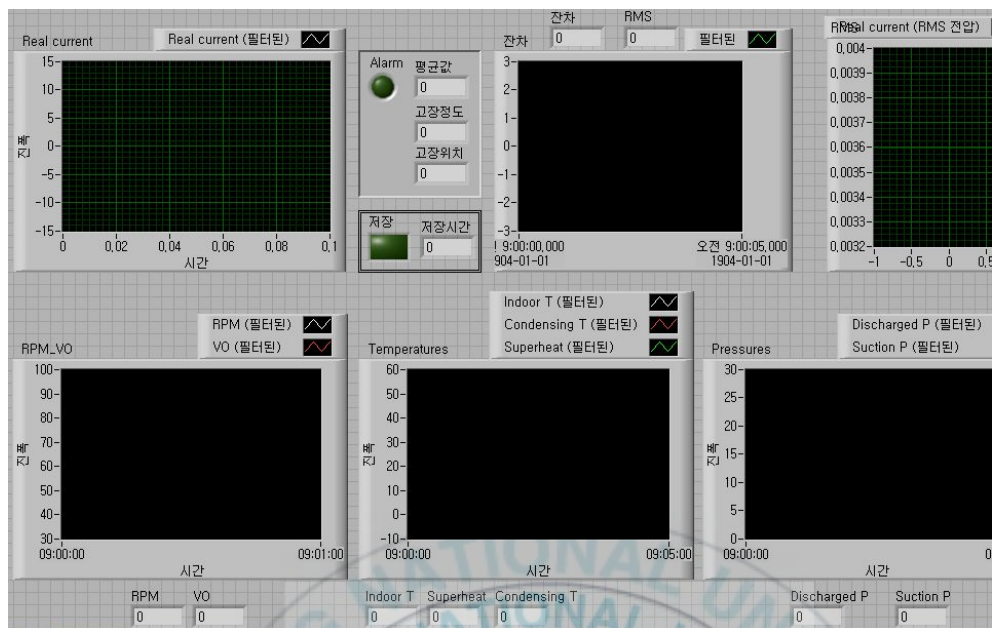
E-1. 제 3장 관련 소스

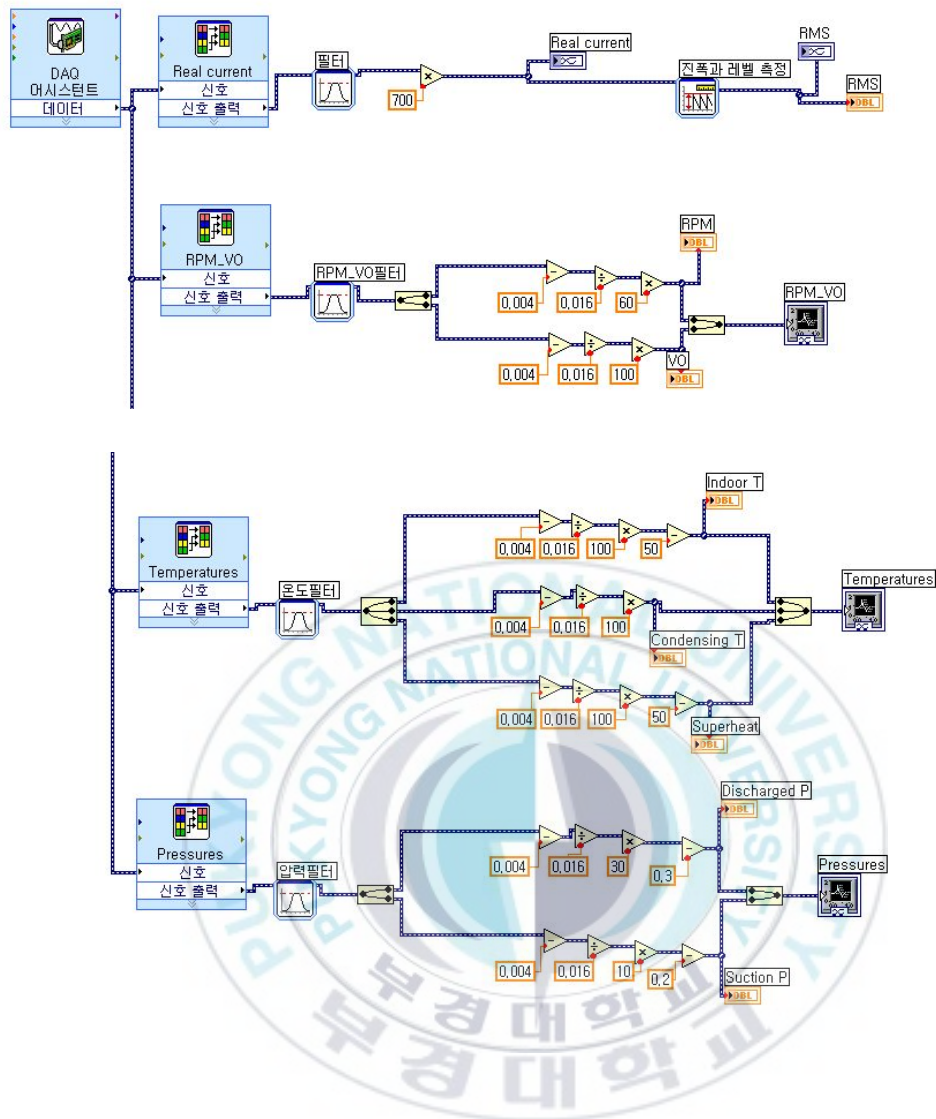


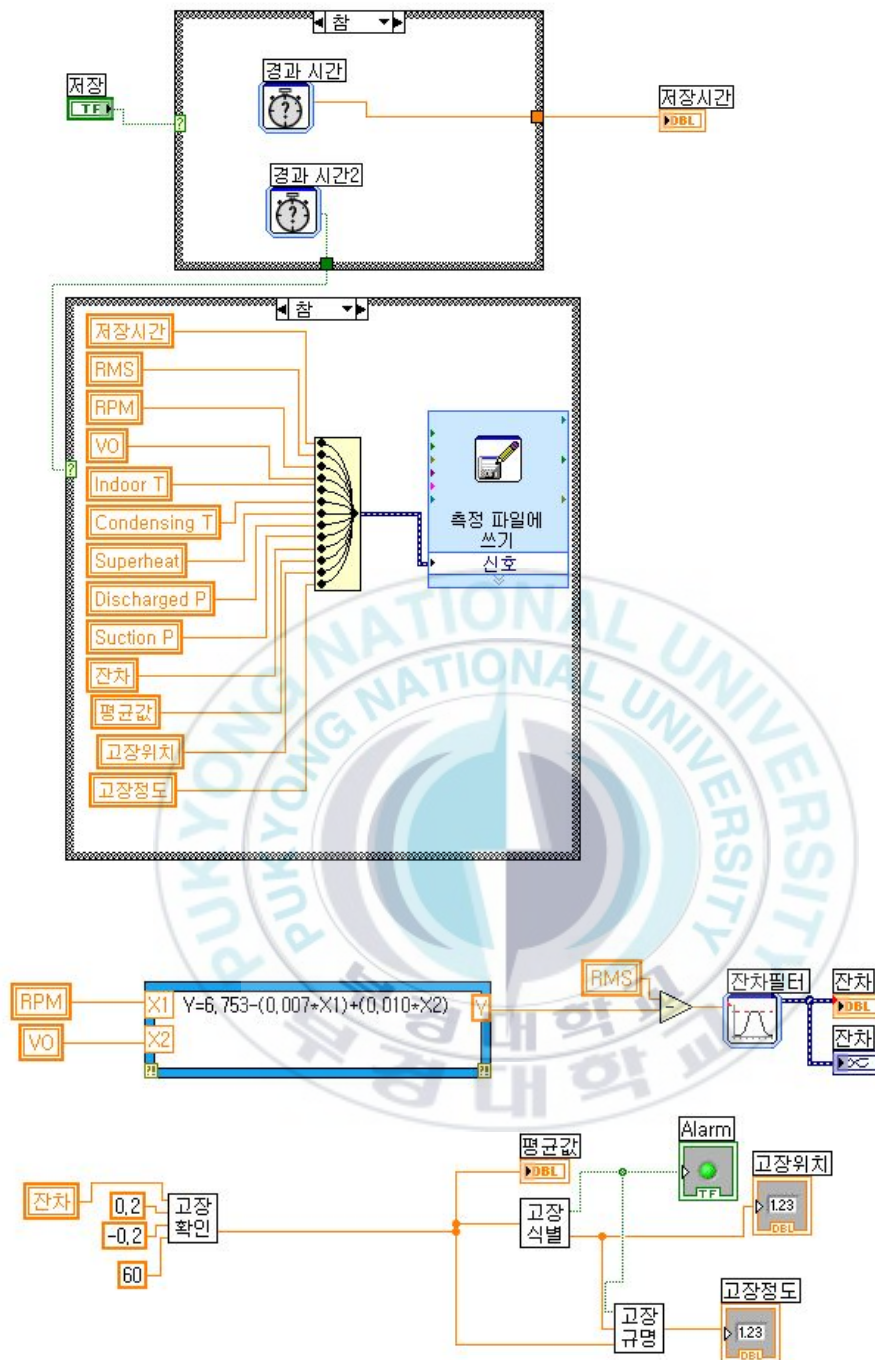




E-2. 제 4장 소스







※ Sub vi의 소스는 지면관계상 첨부하지 않음

감사의 글

AMCL 실험실에 입실할 때가 었그제 같은데 벌써 졸업할 때가 되었습니다. 그동안의 일을 회상하니 즐거웠던 일, 슬펐던 일, 힘들었던 일등이 주마등처럼 스쳐지나 갑니다. 저의 희노애락이 실험실에 배어있고, 저 또한 그 희노애락을 잊을 수 없을만큼 저에게 실험실 생활은 많은 것을 안겨 주었습니다.

그동안 고마우신 분들이 너무도 많지만 가장 고마우신 분은 지도교수이신 정석권교수님입니다. 부족한 저를 받아주시고, 많은 지도와 질타 그리고 격려를 해주시며 자식같이 대해주신 정석권교수님께 진심으로 머리 숙여 감사드립니다. 교수님은 저에게 제 2의 아버지와 같은 만큼 저는 많은 은혜를 입었습니다. 다시 한 번 진심으로 감사드립니다. 그리고 이번 졸업논문의 심사를 맡아주시고 많은 가르침과 조언을 주신 윤정인 교수님과 김종수 교수님, 물심양면으로 도움을 주신 고대권 교수님께 진심으로 감사드립니다.

실험실의 기둥이라 할 만큼 AMCL에 없어서는 안 될 이쁘고 따뜻한 신 이화 선배님, 3년 반동안 같이 동고동락하며 형제같았던 옛실장님이신 이동우 선배님, 저를 AMCL에 입실하게 하고 프로그래밍을 처음 접할 때 상세하게 가르쳐주고 취미가 비슷하여 많은 여가시간을 같이 보냈던 윤종수 선배님, 그리고 동아리 시절부터 같이 공부하고 실험실에도 입실하여 같이 공부하면서 희노애락을 같이 한 둘도 없는 선배님 최정필 선배님께 진심으로 감사드립니다. 학업에 대한 열정으로 메카트로닉스 협동과정에 입학해서 AMCL의 멤버가 되신 정태영, 김종백 선배님, 연구실에서 열심히 공부하고 있는 학부생 안세진, 안성훈, 안성현, 오태영, 이태열, 이동훈 후배님, 졸업하셔서 중국에서 부교수로 계신 이진국 선배님, 취업하셔서 열심히 일하시고 계신 김성하 선배님, 연두흠 선배님, 한유빈 양, 정준우 군 외 AMCL에 몸담았었던 모든 이들에게 깊은 고마

움을 전하면서 앞날에 무궁한 발전이 있기를 기원합니다.

교수님과 여러 선후배님들이 있었기에 지금 제가 여기 있는 것일 겁니다. 부족하지만 저를 이끌어주시고, 믿어주시고, 잘못되면 질타해주신 모든 분들께 감사의 글을 올립니다.

그리고 저의 고뇌를 항상 말없이 들어주고, 고민해주던 대학친구들과 집현전학우회 여러분들께 감사드리고, 고등학교 때부터 우정을 함께 나누며 지금껏 같이 있으면 부러울 것이 없는 FC Magma 친구들에게도 감사드립니다.

마지막으로 저로 인해 많은 고생을 마다하지 않으신 아버지, 어머님께 이 글을 빌려 그동안 하지 못했던 “사랑합니다.” 라는 말을 전하고 싶습니다. 그리고 말은 안하지만 항상 저를 지켜주시던 친형에게 미안하고 고마운 마음을 전하고 싶습니다.

이제 사회에 진출하여 하얀 눈발에 저의 발자취를 하나 세기려 합니다. 저의 발자취가 거뒀되지 않고, 올곧게 앞으로 나가기 위해서 많은 고마우신 분들의 조언을 가슴속에 하나하나 각인시키겠습니다. 그리고 나의 앞 길보다는 주변을 돌아볼 줄 아는 그런 사람이 되겠습니다. 그리고 지금까지 저를 있게 해준 모든 분들께 진심으로 감사의 마음을 전합니다.

2008년 1월 李 東 奎 拜上