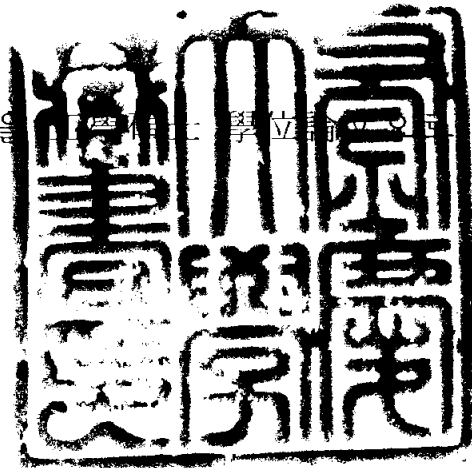


工學碩士 學位論文

포트블 전자 쏙뽀기의 콘트롤러 개발

指導教授 裴 鍾 一

이 論文을 工學碩士 學位論文으로 提出함



2004 年 8 月

釜慶大學校 産業大學院

電氣工學科

金 燾

이 論文을 金 燾의 工學碩士
學位論文으로 認准함

2004 年 6 月

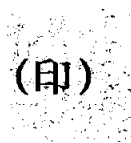
主 審 工學博士 李 東 喆



委 員 工學博士 趙 峰 寬



委 員 工學博士 裴 鍾 一



목 차

그림목차	iii
표목차	iv
Abstract	v
제 1장 서론	1
1. 1 연구배경	1
1. 2 연구목적 및 필요성	3
제 2장 전자 쑥뜸기	5
2. 1 전자쑥뜸기의 개요 및 특징	5
2. 1. 1 특징 및 개요	5
2. 1. 2 약 쑥 엑스제 조제 방법	6
2. 2 PTC 세라믹 발열체의 온도 특성	9
2. 3 임상 치료와 구성 원리	13
2. 3. 1 온열단자	13
2. 3. 2 쑥뜸 패드	14
2. 3. 3 쑥뜸 로션(액쑥 엑스제)	14
제 3장 쑥뜸의 연 역학적 분석	15
3. 1 쑥뜸의 효과	15
3. 2 쑥뜸의 시술 방법 및 분류	16
3. 3 온열의 효과	17

3. 4 피부내장반사이론	19
제 4장 기기 구성 및 쏙뚝기 회로 구현	21
4. 1 PC 사용자의 신체 장애	21
4. 2 포트블 전자 쏙뚝기의 기기 구성	23
4. 3 쏙뚝기의 내부회로 구현	25
4. 3. 1 메인 콘트롤러	25
4. 3. 2 디스 플레이 회로	31
4. 3. 3 전원회로	31
4. 3. 4 발진회로	31
4. 3. 5 리셋회로	32
제 5장 내부 프로그래밍(Software)	34
5. 1 메모리 구조	35
5. 2 ISP(In System Programmer)	36
5. 3 사용 소프트웨어	37
5. 4 전원부 및 콘트롤 회로 구성도	38
5. 5 회로의 기본설정 및 신호동작	39
제 6장 결론	42
참고문헌	43
부록	45

그림 목 차

그림1 쑥엑스제의 조제방법	8
그림2 PTC 저항체의 특성	11
그림3 PTC 세라믹 발열체의 온도특성	12
그림4 온열단자와 쑥뜸패드의 쑥뜸 원리	13
그림5 포트블 전자 쑥뜸기의 블록다이어그램	24
그림6 온열단자와 쑥뜸패드의 구조도	25
그림7 AT90S4433 핀 구성도	28
그림8 AT90S4433의 블록 다이어그램	30
그림9 크리스털 오실레이터의 접속도	32
그림10 MCU 스타트업 빠른 상승 VCC와 RESET 연결	32
그림11 MCU 스타트업 외부적으로 RESET 제어	33
그림12 메모리의 구조	35
그림13 포트블 전자 쑥뜸기의 회로도	38
그림14 포트블 쑥뜸기 사진	41
그림15 포트블 쑥뜸기의 내부 사진	41

표 목 차

표1 PTC 발열체의 구성 성분	10
표2 AT90S4433 마이크로 콘트롤러의 사양 및 주변 장치표	27
표3 AT90S4433 마이크로 콘트롤러의 I/O 핀(Port B) 기능표	28
표4 AT90S4433 마이크로 콘트롤러의 I/O 핀(Port D) 기능표	29
표5 AT90S4433 마이크로 콘트롤러의 기타 기능 핀의 기능표	29

Development of the Portable Moxibustion Caulerizer Controller

Do Kim

Department of Electrical Engineering

Graduate School of Industry

Pukyong National University

ABSTRACT

Electronic moxibustion cauterizer comes with power wealth and a moxibustion mainframe generation of heat pad and is becoming basic configuration with drug pad.

All various products were developed in a use form of the power or a pad form of a SiSul part after it was developed by Pukyong National University JoBongGwan professor, and the electronic moxibustion became a commercialization.

The existing computer loading older brother used only the internal power of PC, but did research and development with portability in order to do PC moxibustion for business use while using the external power through this study.

It is for it is convenient, and characteristics of moxibustion cauterize hold temperature control with a PTC ceramic heating element, and a tube contains a moxa-essence lotion, and it is clean, and a maintenance to have been able to do a component of a medicine for a long term in custody and the hygienic side.

While PC work rulers do business by this study, far-infrared and a component of a mugwort promote the blood circulation in neck, a shoulder, an arm, pain back time for symptoms to be comparatively slight of a wrist part and a preventive dimension,

It was done a nutritious component during oxygen in order to reach by the peripheral nervous system and oils and fats always did health, but did research and development in order to become help.

제 1장 서 론

사람은 누구나 인간다운 삶을 살아가기를 원한다.

그러기 위해서 평소에도 건강에 관심을 가지고 나름대로 관리하고 있으나 운동부족, 영양과잉, 스트레스, 환경오염, 비만 등의 건강한 삶을 저해하는 요인이 증가하고 있다.

건강은 신체의 모든 기관들이 정상의 상태에서 그 기능을 함으로서 주변 환경의 변화에 따라 정상적인 반응을 하는 것이다.

현대에있어 과학과 기술의 발전에 따라 사람은 1차산업에서 3차산업으로의 작업환경의 변화를 가져 왔다.

그 결과 좁은 사무공간에서의 업무시간이 증가하였고 컴퓨터의 보급으로 각자의 근무위치에서 이동하지 않아도 PC를 통하여 모든 업무를 할 수 있게 되었다. 업무로 사용하는 컴퓨터를 계속 사용하면서 전자쑥뜸기를 보다 효과적으로 이용하기 위하여 새로운 컨트롤러를 연구하고자 한다.

1. 1 연구 배경

전통의학으로 분류되는 각종의 의학관련 책자에 따르면 . “관원혈에 1년에 뜸 300 장석을 뜬다면 그 효과가 매우 좋아서 약하던 몸이 튼튼해지고 질병이 없어지니 어찌 수명 연장이 되지 않겠는가” 라거나 삼리혈(三里穴 : 팔에는 수 삼리, 다리에는 족 삼리가 있다.)에 “계속 뜸을 뜨면 전염성 질환이 침범하지 못할 뿐 아니라 매우 건강하게 지낼 수 있다”는 기록이 있다.

이처럼 오랜 역사를 지닌 뜸은 지속적으로 민중의 사랑을 받으며 일상

생활 속에서 널리 활용 되어왔으나 현대 의학의 발전과 서양의학의 도입으로 인하여 점차 잊혀져 갔다

그 이유는 뜨거움이라는 고통을 받지 않고 간편하게 병을 치료받을 수 있는 치료 방법의 등장이고 쑥뜸을 이용한 방법은 의학적 메카니즘을 확인할 수 없어 질병을 치료하는 과정이나 그 효과가 단지 “냉기를 몰아낸다”거나 쑥 기운이 경락을 따라 운행하면서 “약효를 발휘한다”는 등의 합리적이지 못한 설명뿐이라는 것이다.

쑥뜸이란 한 마디로 말린 쑥을 잘 뭉쳐서 몸에다 올려놓고 불을 붙여 말린 쑥이 서서히 타는 동안 그 열 자극이 체 표면에 전달되도록 하는 민간요법이다. 피부 위에서 말린 쑥을 태움으로써 약 60°정도의 온도로 가벼운 화상을 주는 등의 자극을 줌으로서 우리 신체 내에서 발생하는 특수한 단백질(蛋白質)의 화학작용에 의하여 몸 전체가 병에 대한 저항력이 생기고 신체의 균형을 이루게 하는 것을 원리로 삼고 있다.

민간요법적 차원의 요법이란 잘못 되었을 때의 부작용으로 약간의 화상을 입는 물질이 생기는 정도 일 것이다. 하지만 이미 특별한 증상으로 고통받는 환자에게는 심도있는 지식과 노련한 임상적 경험의 바탕이 필요한 경우는 이미 의료 행위가 된다. 또 뜸을 얼굴이나 머리부위 심장 부위에 처방을 할 때 전문가의 도움이 절대적으로 필요하기 때문에 한의학적 치료행위가 된다.

이렇듯 쑥에 불을 붙이는 쑥뜸은 열기는 강하지만 오랫동안 열을 지속적으로 전달하지 못하므로 지속적으로 일정온도를 공급하는 전자쑥뜸기의 회로구성에 대하여 재조명하여 새로운 회로를 구성해 보고자 한다.

1. 2 연구목적 및 필요성

인류가 살아가면서 농사나 상품의 거래 과정에는 숫자의 계산이 필요했다. 요즈음은 여러 가지의 도구가 계산에 이용되지만 옛날의 초기에는 간단히 셈을 필요로 하는 수판을 많이 사용하였다. 수판은 기원전부터 중국이나 이집트, 그리스, 로마 등에서 사용하여 왔다.

1642년 프랑스의 수학자 파스칼이 19세의 나이에 톱니바퀴를 이용하는 최초의 가감기를 만들었고 1671년 독일의 라이프니쯔는 파스칼의 계산기를 개조하여 곱셈과 나눗셈을 할 수 있는 구조로 만들었고 1946년에는 미국 펜실베이니아 대학 교수인 에커트와 머큐리에 의해 세계 최초의 전자 계산기인 에니악(ENIAC)이 만들어 졌다.

컴퓨터는 초기 수 만개의 진공관과 10톤의 무게, 길이 30미터나 되는 크기였으나 그 후 트랜지스터 방식과 3세대 집적회로(IC)를 거쳐 4세대 고밀도 집적회로(LSI)가 사용되기 시작하여 더 빠르고 더 작아지고 더 많은 양의 자료를 기억하는 컴퓨터를 사람들은 사용하게 되었다.

이제 컴퓨터는 고도기술사회의 생활의 주된 매개체로서 사회의 각 분야에서 활용되어지고 이로서 사무환경에도 많은 변화를 가져왔다.

사람들은 PC 앞에서 PC를 이용하여 하루의 업무를 열게 되었고 PC를 통하여 업무를 마감하게 되었다. PC로 인하여 업무능률은 혁신되었지만 인간은 보다 짧은 시간에 업무효율을 극대화하기에 이르렀고 이로 인하여 스트레스도 증가하였다. 컴퓨터 증후군(VDT)이라는 새로운 병들이 생기고 장시간 일정한 자세를 유지함으로써 몸이 경직되고 각 부위의 저림 현상 등 운동부족과 전자파에 노출되어 피로를 가중시키는 요인을 발생 시켰다.

사람들은 경제적 여유가 있자 단순히 살아가는 것보다 삶의 질에 대한 관심이 높아지고 있다. 그래서 사람은 자기의 인생을 “건강하게, 보다 길

게, 보다 풍요롭게” 살기를 원하고 있다.

건강은 허약하지 않거나 병에 걸려 있지 않다는 것만이 아니고 육체적, 정신적, 사회적으로 신체를 완전한 상태로 유지하는 것이다. 사람은 누구나 건강하게 살기를 원하고 있다. 건강을 위하여 필요한 운동을 하여야 하나 현대 사회에서는 문명의 발달로 생활 운동과 자연 운동이 점점 줄어들고 있기 때문에 정상적인 삶은 영위하고 삶의 질을 높이기 위해서는 건강해져야 하고 의식적으로 노력하지 않으면 안 되게 되었다.

현대는 인간의 사고력이 노동력보다 가치를 더 인정받기 때문에 앉아서 하는 일(sedentary life)에 시간을 보내는 사람이 많아졌다. 그래서 인간은 생리기능을 정상적으로 유지시키기에 충분한 신체 활동이 불가능해졌다. 정보화 사회를 살아가는 요즈음, 사람들은 컴퓨터를 사용하여 모든 업무를 하게 되었고 많은 시간을 앉아 있음으로서 혈액순환 장애로 인해서 어깨, 목, 팔, 다리에 근육 계통의 증상들이 늘어나 고통에 시달리게 되었다. 이러한 문제점, 비교적 가벼운 증상들의 완화와 치료를 목적으로 컴퓨터 업무도 보면서 건강을 유지 할 수 있도록 휴대용 전자쭉뽀기를 구현하게 되었다. 전자쭉뽀기는 약쭉으로부터 혈관 확장의 효능이 있는 성분과 황산화 작용을 하는 성분을 추출하고 시술부위에 피부화상을 방지하면서 압의 충격을 억제한다는 43[℃] 정도의 최적온열을 공급하는 세라믹 발열체를 이용하여 화상 및 화재의 위험성을 제거하고 시술의 안전성과 간편성이 추구 되도록 설계 고안되었다.

명량하고 밝은 사부분위기 조성과 컴퓨터를 이용하는 사람들의 가벼운 고통을 덜어주고 컴퓨터 업무를 하면서 간단하게 조작 시술할 수 있는 전자쭉뽀기의 컨트롤러를 연구 개발하고자 한다.

제 2장 전자쑥뜸기

2. 1 전자쑥뜸기의 개요 및 특성

2. 1. 1 특징 및 개요.

전자쑥뜸기는 기존의 전통 쑥뜸 시술시 사용되는 약쑥 봉치 대신에 쑥뜸을 사술할 부위에 쑥 엑스제를 도포하여 약쑥의 약리작용을 간편하고 위생적으로 사용할 수 있는 쑥엑스제를 약쑥으로부터 추출하여 조제하고 쑥뜸을 시술할 수 있도록 하기 위한 것이다.

약쑥 앞에는 쑥 디클로로메탄 ($\text{Moxa} - \text{CH}_2\text{Cl}_2$)과 에틸 아세테이트 ($\text{Moxa} - \text{EtOAc}$) 성분이 있는데 이 것을 각각 추출하여 쑥엑스제를 조제하는 방법과 추출된 쑥 디클로로 메탄과 에틸 아세테이트로부터 쑥 엑스 톱크제나 쑥 엑스 로션제를 조제하는 방법과 전극단자 두 개를 갖춘 전극으로 공급되는 전류가 일정한 온도를 발열하는 장치로서, 전극 두 단자와 연결되어 발열장치에 외부 전원을 공급하는 전원공급장치와 발열장치의 상부와 측면전체를 둘러 감싸는 형태의 발열체 고정단자에 접촉 조립되어 외부로 열이 분산되지 않도록 단열되고 피부에 부착되는 피부 전극 패드로 구성되어 있다.

전자쑥뜸기로 쑥뜸을 하는 방법은 새로운 기술을 이용하는 쑥뜸으로서 약쑥에서 추출한 약쑥 엑스제와 전자 및 신소재 기술을 활용하는 피부전극형 쑥뜸기(PTC발열체) 열단자 패드를 사용하는 것으로 시술자의 편의에 따라 한 개의 전극과 여러 개의 전극을 사용하는 다중 패드형이 있으며 시술 전에 피부에 약쑥 엑스제를 바르고 난 후에 피부 부착형 약쑥패드 전극과 쑥뜸기를 결속한 후 전원을 공급하여 사용한다.

이제까지의 쑥뜸 방식은 시술시 쑥봉치에 매번 불을 붙여서 하는 간접

구 방식으로 시술자가 쑥 연기에 노출되고 시술자의 몸 움직임(기침, 재채기, 팔·다리 위치이동)으로 인하여 화상의 위험도 있었고 쑥 봉치의 호트라짐, 태우고 난 재의 처리 등으로 주변을 청소해야 했으며 시술자가 시술시 고정자세를 취하고 오랫동안 있어야 하는 불편도 있었으나 그나마 한꺼번에 여러 곳을 시술할 수도 없었으나 전자쑥뜸기는 신체의 15 곳까지 한꺼번에 시술이 가능하도록 개발된 것이 특징이다.

2. 1. 2 약쑥 엑스제의 조제방법

약쑥 엑스제는 약 쑥 가공의 3단계 공정을 통하여 조제된다.

가. 약쑥과 메탄올(MeOH)로부터 쑥 메탄올 엑스를 3회 추출한다

ㄱ) 첫 번째 : 용기에 약쑥 애엽을 넣고 여기에 유기 용매인 메탄올을 넣고 밀폐하여 상온에서 24시간 경과시키면 메탄올은 쑥 메탄올이 된다.

(여기서 추출된 쑥 메탄올을 다른 용기에 옮겨 두고)

ㄴ) 두 번째 : 한번 추출된 약쑥 애엽에 메탄올(Me OH)을 다시 밀폐하여 상온에서 다시 한번 24시간을 경과시키면, 메탄올에서 또 다시 쑥 메탄올이 된다. (여기서 추출된 쑥 메탄올을 앞에 보관된 용기에 옮겨서 함께 보관하고)

ㄷ) 세 번째 : 2회 추출된 약쑥 애엽에 메탄올을 다시 넣고 밀폐하여 $70\sim 80[^\circ\text{C}]$ 온도에서 가열하여 3시간을 경과시킨다. 그러면 메탄올은 또 다시 쑥 메탄올이 된다.

앞의 2회 추출된 쑥 메탄올과 세 번째 추출된 메탄올을 함께 섞어 농축시키면 쑥 메탄올 엑스를 만들 수 있게 된다.

나. 쑥 메탄올 엑스로부터 쑥 디클로로 메탄과 쑥 에틸 아세테이트를 분류한다.

가) 농축된 쑥 메탄올 엑스를 물에 현탁시킨 다음에 디클로로 메탄올을 섞어 분획 분취기에서 분획하면 비중차이에 의해서 아래쪽에는 분리된 쑥 디클로로 메탄 추출물을 얻게 되는데 쑥의 정유 성분인 치네올, 알파루존 지방산이 여기에 포함된다.

나) 나머지 남아 있는 성분에 다시 에틸 아세테이트를 섞고 분획 분취기로 다시 분획하게 되면 비중차에 의해서 위쪽 부분에서 쑥 에틸 아세테이트를 분리하여 추출하게 된다.

다) 이 때 쑥 디클로로메탄 성분에는 혈관 확장제(vasodilator)를 쑥 에틸 아세테이트 성분에는 노화 방지제(antioxidant)가 포함된다.

그 다음 최종적으로 남아 있는 부분을 농축시키면 쑥물엑스(쑥의 인, 아미노산, 펩티드 성분)가 된다.

나. 쑥 디클로로 메탄과 쑥 에틸 아세테이트를 혼합하여 약쑥 엑스제를 조제한다.

분획하여 추출한 쑥 디클로로 메탄과 쑥 에틸아세테이트 그리고 쑥물엑스를 혼합하고 농축하여 물파스처럼 피부에 도포 하여도 사용이 용이하도록 메탄올을 3~10배로 희석하여 쑥 엑스제 완성품을 조제한다[1].

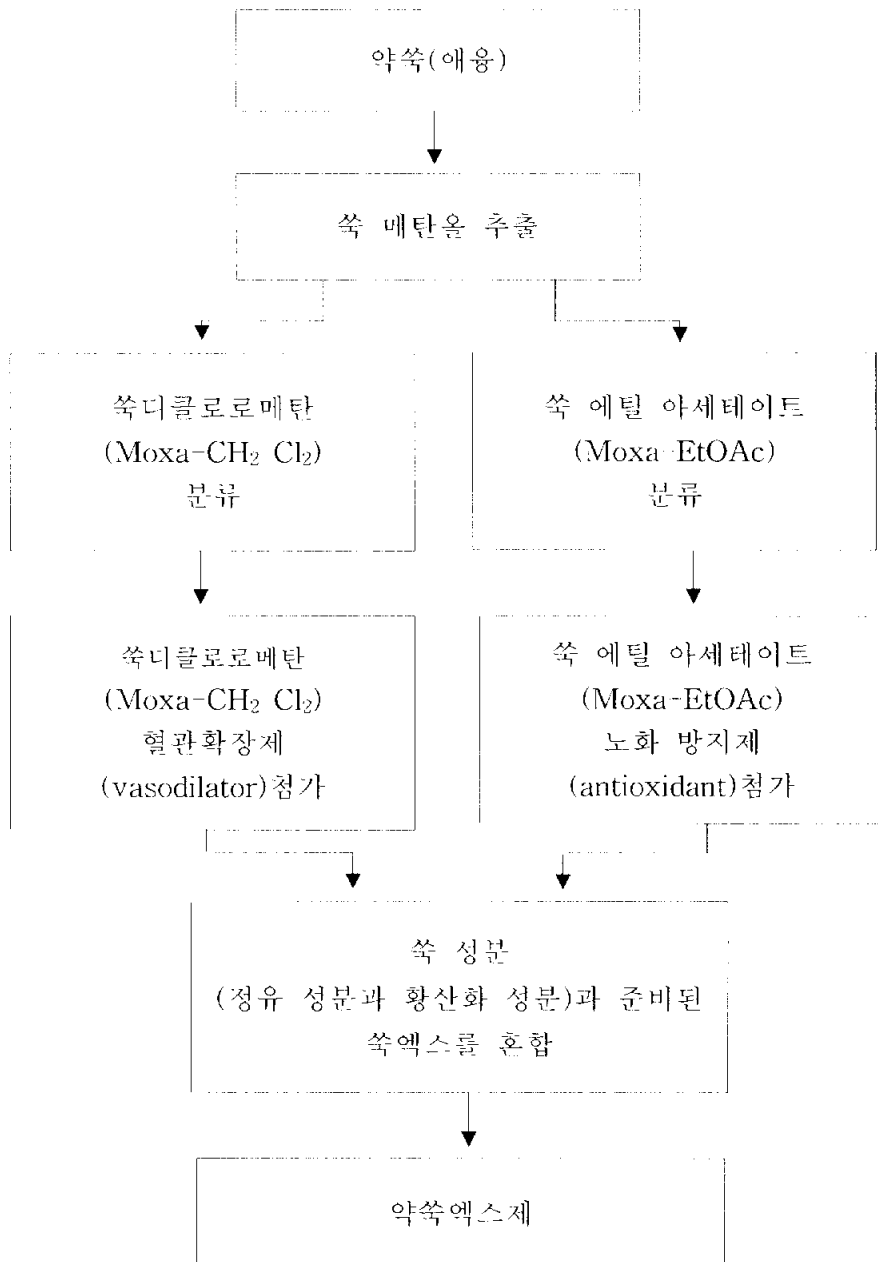


그림 1 약쑥엑스제의 제조방법

2. 2 PTC 세라믹 발열체의 온도 특성

PTC THERMISTOR는 (Positive Temperature Coefficient thermistor) 정온도계수를 말한다. 즉 PTC 서미스터는 온도가 상승함에 따라 저항이 증가하는 특성을 갖춘 thermistor로서 정온도 계수를 말한다.

특성을 보면,

가. 낮은 온도에서는 비교적 작은 저항치를 갖는다.

나. 일정 온도에 도달하면 이 온도부터 급격하게 저항이 증가하고 그 증가폭은 대단히 크게 된다.

다. 전류가 통하면 일정온도 이상에서 전류가 흐르기 어렵게 된다.

위 특성과 같이 성질이 급변하는 온도를 「퀴리 온도」 또는 「퀴리 점」이라 하며 온도가 이를 넘어서면 정 특성의 저항 급증 영역에 들어가게 된다. 세라믹스의 성분을 변화시켜 특성을 바꾸는 방법은 다음과 같다.

특성을 바꾸는 주성분은 티탄산바륨(Ba_2TiO_4)이다. 바륨(Ba)의 성분 일부를 납(Pb)으로 바꾸면 퀴리 점은 고온 방향으로 이동하게 되고, 바륨을 스트론튬(Sr)으로 치환 또는 티탄(Ti)의 일부 성분을 주석(Sn) 이나 지르코늄(Zr)으로 치환하게 되면 퀴리 점은 저온 쪽으로 이동하게 된다. 이와 같이 실험하면 그림 2와 같은 여러 가지의 퀴리온도를 측정할 PTC 서미스트가 만들어진다. 그림 2에서 보듯이 PTC의 온도 저항곡선이 합금성분에 따라서 여러 형태로 움직임을 알 수 있다.

이렇게 해서 목적에 맞는 퀴리점과 티탄산 바륨계 세라믹스 원판을 만들고 그 양면에 전류를 통과시키기 위한 전극을 밀착시켜야 한다.

PTC 서미스트는 온도의 변화에 민감하게 저항이 증가하는 소자 라는 점에서 온도센스, 온도로 전류를 제한하는 소자 또는 스스로가 저항체이므로 과도한 전류가 흐르면 발열하고 이 온도 상승으로 저항이 급증, 전류를 억제하는 과전류 보호소자 등 여러 가지 목적으로 사용되어 많은

특성의 제품들이 사용되고 있다. PTC 세라믹((Ba 0.8 Sr 0.2) 0.996 Y0.004 TiO₃ + 0.5WT SiO₂ [%])의 측정 결과는 그림3과 같다. 측정 방법은 PTC 세라믹(Φ12×두께1.2[mm])의 경우에 DC전원을 2[V] 단위로 30분 간격으로 전류와 온도를 측정하였다.

측정한 결과 23[℃]일 경우 24[mW]의 소모 전력량이 생겼고, 31.5[℃]인 경우 100[mW]의 소모량, 40[℃]의 경우는 소모량이 200[mW], 그리고 50[℃]의 경우 300[mW], 55[℃]의 경우 400[mW]의 소모 전력량이 됨을 알 수 있다. 그러므로 최소의 전력량으로 온열 시술을 할 수 있다.

그림3에서 보듯이 쿨리 포인트는 55[℃]에서 형성되며 전자쑥뜸기에서 요구되는 중요한 온도의 특성인 40~60[℃]의 온도를 일정한 DC의 전압 공급으로 온열 특성을 얻을 수 있다.

표 1 PTC 발열체의 구성 성분

성 분	분자량	분자당 계산량
BaCO ₃	197.35	BaCO ₃ 의 분자량 × 0.8 × 0.996
SrCO ₃	147.6	SrCO ₃ 의 분자량 × 0.2 × 0.996
Y ₂ O ₃	225.81	Y ₂ O ₃ 의 분자량 × 0.004
TiO ₃	79.9	TiO ₃ 의 분자량
SiO ₂	.	위의 총량 × 0.5 × 0.01

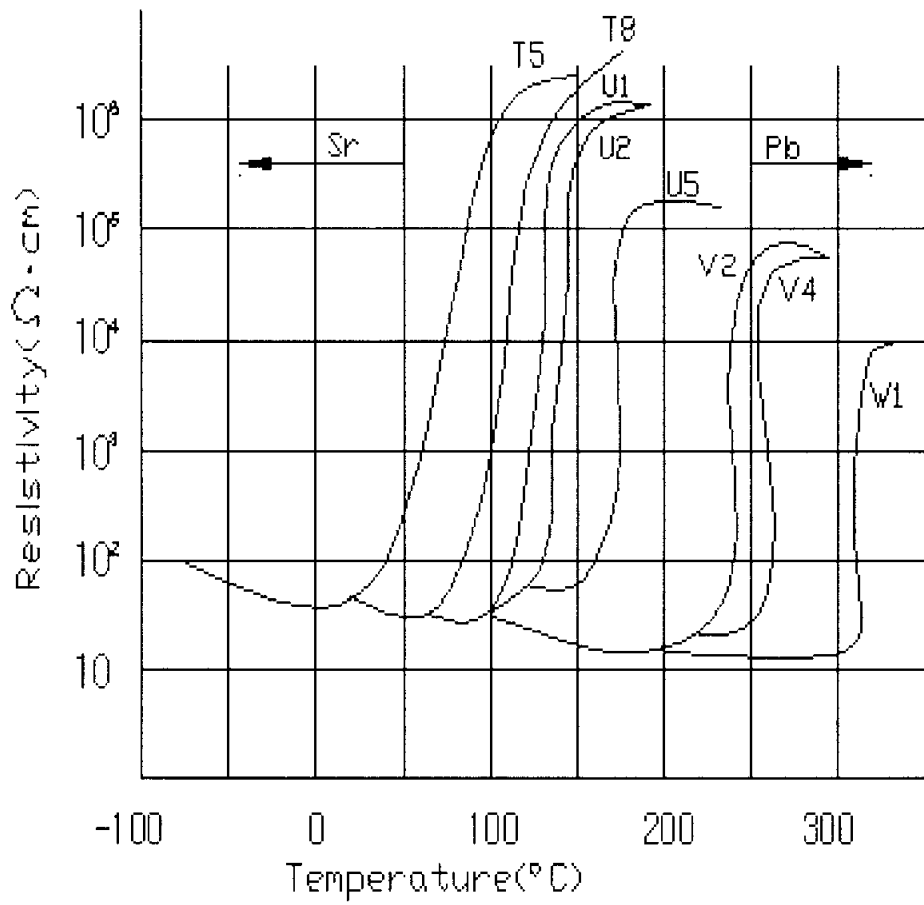


그림 2 PTC 저항체의 특성

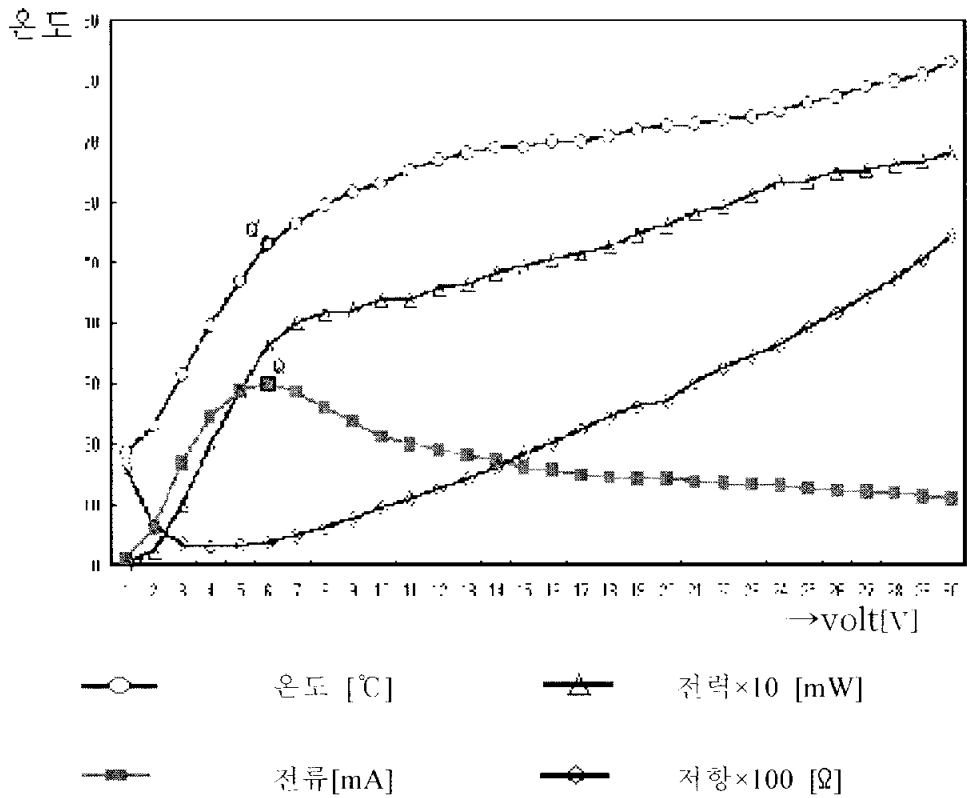


그림 3 PTC 세라믹 발열체의 온도특성

그림 3의 PTC 세라믹 발열체의 온도특성 그래프에서 나타내어진 것처럼 저항의 특성은 20[°C]에서 가장 낮으며 점차적으로 온도가 증가하면서 저항도 점차적으로 증가함을 알 수 있다.

퀴리포인트의 지점이란 전류가 가장 많아지는 Q지점에서의 온도 Q'는 55[°C]에서 퀴리 포인트가 얼어짐을 알 수 있다.

2. 3 임상 치료와 구성 원리

약쑥엑스제의 약리 성분이 피부 깊숙이 침투되도록 쑥뜸패드를 만들어 경혈부에 부착시키고(증상에 따라 차이를 둠) 그 위에 PTC 발열체인 온열단자를 통하여 원적외선과 온열이 피하심부에 전달되도록 하여 내분비계, 면역계, 자율 신경계의 조직과 환경을 개선하여 면역력과 자연 치유력이 회복되게 한다[2].

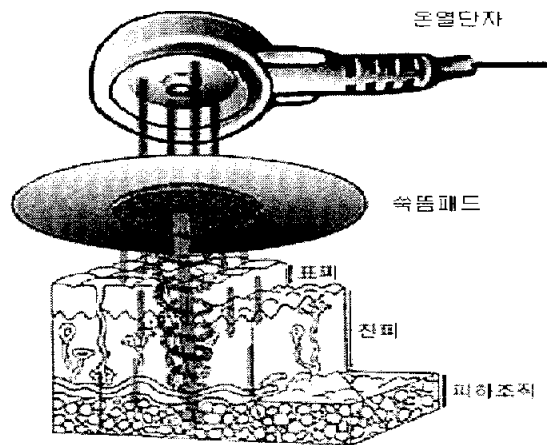


그림 4 온열단자와 쑥뜸패드의 쑥뜸 원리

2. 3. 1 온열 단자(온도 발열 장치)

PTC 세라믹 발열체는 일정한 온도를 일정한 전압에서 항상 유지시키기 때문에 온도 감지용 센스는 필요로 하지 않는다. 그래서 온열 단자의 부피를 최소화, 경량화할 수 있다. 이는 기존과 전자쑥뜸기(조막사 개발 이전의 제품)와 대별되는 새로운 방식이다. 지금까지의 온열단자는 발열 장치에 온도측정용 센서를 별도로 장착하여 온도를 감지하는 방법으로서

일정온도에서 전류의 흐름을 제어하는 아원된 방식이었다.

이와 같은 방법은 피부에 온열을 공급하는 발열부와 발열부의 온도를 피부측에서 감지하는 센서가 일체형으로 되어야 하므로 온열단자를 경량화 하여도 부피가 커서 불편하였다. 단자가 커지면 피부에 접촉시 접촉재의 부착력이 더욱 강해져야 하고 피부에 가벼운 상처를 줄 수도 있다. PTC 세라믹 발열체는 이 점을 보완하고 개선하여 발열부의 전체 두께가 6mm 정도로서 아주 얇고 가볍다. 이 패드는 신체와 어느 부위에 붙어도 부착력이 좋으며 분리하지 않는 일상의 업무는 하여도 문제가 없는, 활동성이 아주 좋은 것이 특징이다. 그래서 일정한 자세를 유지만 한다면 데스크 작업이나 PC 등의 업무는 하여도 무방한 것이다.

2. 3. 2 쑥뜸패드

경혈점은 경혈점이 아닌 피부 부위에 비해서 전기가 잘 흐르는가? 그리고 중풍으로 한 쪽에 마비가 있을 때, 마비가 있는 쪽과 거동이 괜찮은 쪽 사이에 전기적 특성은 어떠한가? 또한 폐, 비, 간, 신 관련 장기의 만성병 환자를 대상으로 폐, 비, 간, 신의 윤�혈(兪穴)과 모혈(募穴) 사이에 전기적 특성상, 눈에 뵈는 만한 차이가 있는가? 이런 연구를 하면서 피부로부터 전기적 신호를 감지 해주는 근전도(筋電圖) 패드를 사용하게 되었다. 근전도 패드는 근육으로부터, 심전도 패드는 심장으로부터, 미세한 생체 신호도 놓치지 않고 잘 감지해 주는 구조로 되어 있다. 이러한 패드에는 전해질이 들어 있는데 이 전해질은 피부의 각질층에 스며들게 되면 전기가 잘 통하게 한다. 이 전해질이 바로 쑥엑스제로 바뀌어져서 패드 안에 들어가게 된 것이 바로 약쑥패드이다.

2. 3. 3 쑥뜸로션(약쑥엑스제)

약쑥으로부터 추출한 약쑥엑스제에는 혈관 확장 효능이 있는 정유 성분과 피부 노화를 억제하는 항산화 성분 등이 함유되어 있다[3].

제 3장 쑥뜸의 열역학적 분석

3. 1 쑥뜸의 효과

쑥뜸은 화상의 위험성과 시술의 비과학성 등의 문제점을 안고 있었다. 따라서 이런 점들을 극복하고 과학화하려는 연구가 진행되었고 쑥으로부터 혈관확장성분과 피부노화를 억제하는 항산화 성분이 함유된 쑥엑스제를 만들어 일정온도와 원적외선을 쏘여 시술하는 단계에 이르렀다[4].

3. 1. 1 억제 작용

뜸으로 인체에 강한 자극을 가하면 실한 것을 사해내는 사법을 쓰면 진통, 진정, 제지 등의 효과가 있다. 효과는 크게 세 가지로 다음과 같다.

가. 지각 신경이 과민반응으로 이상 흥분해져서 등통이 생긴 경우에 이 것을 진정 시킬 수 있다.

나. 운동 신경에 이상 흥분이 생겨서 경련이나 마비 등의 증상이 나타날 경우 이를 억제시킨다.

다. 자율신경이 흥분하여 해당기관과 조직의 기능에 이상이 생길 때 이를 억제하여 정상화 할 수 있다. 즉 항진을 억제시킬 수 있다.

3. 1. 2 흥분 작용

흥분 작용은 뜸의 보법으로 뜸으로 신체에 자극을 가하면 각 신경(지각, 운동, 자율)계에 기능이 저하되거나 쇠퇴하였을 때 신경을 관장하는 해당 기관에 조직의 기능을 흥분시켜 활성화 되게 할 수 있다.

3. 1. 3 유도 작용

뜸 치료법은 아픈 부위에 직접 자극하지 않고 아픈 부위보다 멀리 떨어진 곳에서 자극을 주어도 혈관을 확장시키거나 수축되게 한다. 이로서 혈액의 순환과 병적 삼출물의 대사와 배설을 촉진시키게 된다.

3. 1. 4 반사 작용

오장육부나 혈관 내분배선 등의 각 기관에도 피부의 일정 부위에 뜸으로 자극하면 반사적인 영향을 주어 질병의 치료에 도움이 된다.

3. 1. 5 면역 작용

뜸을 뜨면 신체 내에 이중 단백질이 생성되어 항체가 강화되어 면역작용이 생긴다, 병원균이나 독소류가 몸 안에 들어와도 ‘리오나카신’ 등의 물질이 생성되어 백혈구의 식균작용을 왕성하게 한다.

3. 1. 6 증혈작용

뜸을 뜨면 적혈구 및 혈색소가 크게 증가한다. 혈액의 증가는 피의 순환을 활발하게 하고 산소 등의 영양소를 몸의 말단부까지 신속하게 보낼 수 있어서 혈색소가 폐에서 취한 공기중의 산소와 결합하여 동맥혈인 신선한 피를 말초 조직까지 보내게 되므로 인체를 더욱더 건강하게 한다.

3. 2 쑥뜸의 시술방법 및 분류

3. 2. 1 직접 쑥뜸

직접구라고하며 뜸 기둥을 혈자리 위에 직접 놓고 태우기 때문에 화상으로 상처가 남게 된다. 직접 뜸 요법은 화상을 입히는 방법으로 동의보감에서 허준의 설명은 “뜸을 떠서 병을 치료하는데 장수를 넉넉히 떼다고 해도 뜸자리가 헐어서 고름이 나지 않으면 효과가 없다”고 하였으며 이 방법은 치료 후에 상처가 남게 되므로 신중을 가려야 할 치료법이다.

3. 2. 2 간접 쑥뜸

간접구 쑥뜸은 혈 자리에 화상을 입히지 않고서 뜸 치료하는 방법이다. 혈 자리에 마늘, 생강, 소금, 부자 등을 깔고 그 위에 뜸 기둥을 놓고 시술한다. 마늘과 생강 등을 얇게 썰어서 구멍을 몇 개 뚫은 후 그 위에

땀을 뚫고 연소시키기 때문에 열의 강도를 많이 경감하게 되며 뜨거울 때는 중단했다가 시술 할 수 있는 편리함이 있다. 그 외에 소금 땀은 신권혈에 땀뜨는 방법으로 중풍으로 실신했을 때나 급성복통, 구토, 설사, 이질 등의 증상에 효과가 있으나 가열되면 소금이 튀어올라서 피부 화상의 우려가 있다. 땀의 효과를 최대화하기 위해서는 열감을 어떻게 하면 더 많이 인체내에 전달하느냐에 따라 달라지고 약쥬의 약리 작용 성분이 인체내에 얼마만큼 스며들게 하느냐가 중요한 것이다.

3. 3 온열의 효과

3. 3. 1 극소적 효과

Abrahamson의 연구에 의하면 습열(moist heat)을 장시간동안 강력하게 적용하면 약 3.4[cm] 즉 근육의 표층까지 침투한다는 사실을 밝혀 주고 있다. 수 치료에서는 전도열이 깊이 침투되지 못하고 피하나 피하 조직에 제한된다. 피부 표층에 열이 전달되면 생리학적 반응들이 일어나는데 국소적 열이 증가 된 혈액순환에 의해 급속히 분산되기 때문이다.

국소적 온열에 의한 가장 명확한 변화는 조홍(redness)으로 이 것은 혈관의 확장으로 피부 수용기의 자극에 의해 국소 축삭 반사(local axon reflex)에 의해 일어나며, Landis는 국소적 온열이 적용된 부위에서의 모세 혈압관(capillary pressure)이 어떻게 변화되는가를 연구하였는데 모세 혈관압이 사지의 정맥과 동맥 모세혈관에서 현저하게 증가된다는 것을 발견 했다. Abrahamson은 45[℃]의 습열을 전완부(forearm)에 20~30분 가량 적용했을 때 혈류와 율(refe of blood flow)이 2배정도 증가하고 열 적용을 중지 한 후 약 1시간 정도는 계속 되었다고 하였다. 국소적 온열 적용후 그 효과들은 국소 신진대사의 증가(van't Hoff's law)나 온열이 적용된 부위의 혈관을 통한 백혈구의 이동증가, 근육의 이완 그리고 발

한이나 진통효과가 습열에 의해서 나타난다.

3. 3. 2 전신적 효과

온열의 전신 적용은 전신 온습포(full hot packs)나 침수욕(immersion baths), 고온 증기욕(hot vapor baths)에 의해 적용되는데 분명한 생리적 반응이 있다. 이 반응은 체온이 상승되는 것을 저지하려는 인체의 체온 조절기 전에 발생되는데, 말초 혈관이 피부로부터 복사나 대류, 전도, 증발 등에 의한 체열 방출을 증가시키기 위해 확장되고 혈류율의 현저한 증가와 함께 맥박(pulse rate)의 증가가 일어난다. Bierman은 사람에 따라 조금 다르지만 혈류율이 약 400[%]정도 까지 증가 할 수 있다고 하였고 Bazett는 온도 0.6[°C]상승 할 때 마다 분당 10회 정도 맥박이 증가한다고 하였다.

온열 전신적용시의 변화 내용을 보면

가. 혈류율이 약 400[%] 증가된다.

나. 온도 0.6[°C] 상승시 분당 10회 가량 맥박 수가 증가 된다.

다. 40[°C] 이상 습열을 전신적으로 적용 시 심박동율(heart rate)증가하고 심박출량(stroke volume)은 감소한다.

라. 확장기 혈압(diastolic blood pressure)은 하강하고 맥압 (pulsepressure)은 상승한다.

마. 온도가 0.6[°C] 상승할 때에 호흡 율(respiratory rate)이 5~6회 정도 증가한다.

바. 40~41.7[°C] 온열 적용 후 백혈구가 증가하고 온열적용 완료 후 수 시간 지난 다음 백혈구의 수가 최대로 증가(적용전1[mm³])당 71.25, 적용후 1[mm³]당 11270, 수 시간 후 1[mm³] 40,000) 백혈구 수의 증가는 온열을 장시간 적용하면 더욱더 많이 증가하고 체온이 40[°C]에 도달하면 백혈구의 운동이 최대로 증가한다.

습열을 전신에 적용하면 액체가 혈류내로 이동해 가며 혈액의 체적이 전체적으로 다소 증가하고 혈액이 약간 희석되어 1[mm]³당 적혈구 수가 상대적으로 약간 감소하고 온열 치료(fever therapy)에서는 1[mm]³당 세포의 수가 증가되어 오히려 농축이 일어나고 온열을 전신에 적용하면 발한의 증가와 수분, 염분, 그리고 소량의 요소, 요산, 크레아틴, 인산, 황, 젖산 등의 상실이 일어난다[5].

3. 4 피부 내장 반사 이론

피부 표면에 온열이나 한 냉을 강력하게 국소 적용하면 피부 표면에 대한 즉각적인 반응뿐만 아니라 신경 지배를 통하여 신체의 다른 부위에서도 원격 반사(remote reflex)나 혹은 공감성 효과(consensual effects)를 나타낸다.

3. 4. 1 반사 혹은 공감성 효과들

반사 혹은 공감성 효과는 혈관운동 또는 순환에 대한 효과, 내장 운동 또는 근육에 대한 효과, 분비 또는 선(gland)에 대한 3가지 효과로 크게 대별된다.

가. 반사효과

피부와 내장기관 사이의 분절적 관계에 대한 해부학적 기초를 근거로 내장기관으로부터의 강력한 통증이 척수의 같은 분절에 의해 감각지배를 받는 피부 부위에 연관통을 유발하고 감각이 약간 무딘 부위(내장기관)에서 통증이 유발되면 이 통증의 감각이 해부학 적으로 반사의 관계에 있는 좀 더 민감한 부위(피부)에서 느껴진다는 것을 Head는 발견하였다.

나. 내장운동 및 근육 효과

Kuntz는 유용성의 관점에서 볼 때 피부의 자극은 내장반사, 특히 혈관

운동의 변화나 내장기관 중의 근육 구조물의 긴장상태에 변화를 주게 되므로 실제로는 장관(gastrointestinal canal)등에 질병이 있을 때 그와 연관된 부위를 적당한 방법으로 자극하면 매우 유효한 효과가 있다 하였다..

나. 분비효과

외적으로 적용된 온열은 내장의 혈류를 감소시킬 뿐 아니라 장의 운동성과 위에서의 산 분비(acid secretion)를 억제하게 하고 한 냉은 온열과 반대 현상을 나타낸다.

3. 4. 2 장시간 온열 적용시의 반사적 효과

가. 한쪽 사지(extremity)에 장시간 온열을 적용하면 반대 측 사지에서 혈관의 확장이 일어난다.

나. 복벽(abdominal wall)에 장시간 온열을 적용하면 장의 활동과 혈류의 흐름이 감소되고 위산의 분비가 억제된다.(Fisher와 Solomon).

다. 골반(pelvis)에 온열을 장시간 적용하면 골반 기관의 구조물이 이완되고 혈관이 확장되며, 월경성 출혈이 증가된다.

라. 전흉부(precordium)에 온열을 장시간 적용하면 심박동률(heart rate)은 증가되고 반면 심박동력(heart force)이 감소된다. 그리고 혈압은 낮아진다.

마. 흉부(chest)에 습열을 장시간 적용하면 호흡과 담(expectoration)의 배출이 증진된다.

바. 체간(trunk)에 온열을 장시간 적용하면 요관(ureters)이나 담관(bile duct)의 명백한 이완이 일어난다.

사. 배부(back)에 있는 신장부위나 앞쪽의 하복부에 장시간 습열을 가하면 요(urine)의 생성이 증가된다.

제 4장 기기 구성 및 쑥뜸기 회로구현

4. 1 PC 사용자의 신체장애

사무실에서 근무하는 사람들 중에는 항상 목이나 어깨가 무겁고 아프다고 호소하는 사람들이 많아지고 있다. 특히 최근에 오랫동안 앉아서 컴퓨터를 사용하는 일이 늘어나고 앉아서 한 가지 일만 하는 것이 아니라 여러 가지 일을 동시에 하는 경우가 많아 자기도 모르게 어깨나 목의 통증이 생기게 되고 목의 근육이 뻣뻣해 지거나 고개를 숙이고 젖힐 때 아프며, 어깨의 양쪽에 근육이 뭉쳐서 딱딱한 것이 만져지며 항상 피곤해 하고 의욕이 떨어지는 것을 느끼는 경우가 많아 졌다.

4. 1. 1 목의 통증을 유발하는 원인과 통증의 증상들

유발원인

가. 컴퓨터 모니터의 위치와 높이

나. 키보드의 높이

다. 컴퓨터에 입력하는 정보 자료와 모니터의 위치

라. 의자 등받이의 강도 및 자세와의 관계

등으로 턱을 목의 앞쪽에 갖다 대거나 목의 아래 부분이나 등의 윗부분이 앞쪽으로 구부러진 상태에서 계속 목을 받치고 있을 때, 통증이 생길 수 있다.

잘못된 자세에서 오는 증상

가. 두개골 아래쪽에 붙은 근육이 짧아지고 조여서 두통을 유발

나. 관절의 과도한 운동으로 목의 중간 부위에 있는 인대가 긴장되어 통증을 일으키고 목의 신경에 압박을 가하게 된다.

다. 목의 아래 부위와 등 윗부분의 강직은 목 근육이 긴장되고 근육이 약해져서 목 아래와 어깨의 통증을 일으키게 된다.

4. 1. 2 어깨, 팔목, 손의 통증 유발 원인

컴퓨터의 마우스를 많이 사용할수록 손, 팔목, 어깨의 통증이나 붓고 무감각해지거나 이외의 다른 증상으로 고통과 불편은 많아진다.

코펜하겐의 국립직업 건강 연구소의 Chris Jensen 박사는 작업시간의 50% 이상을 컴퓨터를 사용하는 사람에서 손이나 팔목에 문제가 발생할 위험성이 높으며 컴퓨터 앞에서 거의 하루종일 작업하고 적어도 반 이상의 시간을 마우스를 사용하면 같은 시간을 컴퓨터를 사용하더라도 마우스를 4만 사용 하는 사람보다 4배의 위험성이 있다고 하였다. 물론 이들에 있어서 마우스 사용뿐 아니라 반복적인 일을 하는 것도 문제이다.

두 번째 연구에서 일주일에 30시간 이상 마우스를 사용한 사람에서 전완(前腕)의 통증이 8배, 극심한 목 통증은 2배 어깨 통증은 3배 발생할 위험성이 많다고 하였다. Chris Jensen 박사는 작업할 때 키보드나 마우스를 가능한 멀리 하라고 조언하고 있다[6].

4. 1. 3 반복 사용 긴장성 증후군(RSI)

같은 동작을 반복하는 물리적인 운동에 의해 잘 생기는 질환으로 최근에 컴퓨터 게임 마니아나 컴퓨터 작업이 많은 사람에서 많이 생긴다.

영국에서는 연간 10만 명 이상의 컴퓨터를 즐기는 아이들의 15%가 이러한 증상을 호소한다고 한다. 반복 사용 긴장성 손상 증후군(Repetitive strain Injury) 즉 RSI를 일으키는 가장 위험한 요인 중의 하나는 반복적 사용 중에 발생하는 근육, 건초 신경에 대한 혈액순환 장애이다. 고정된 자세로 수축(Static contraction)동안에는 혈액 순환이 거의 중지되며 기존의 산소는 소모되어 없어진다. 컴퓨터를 계속 사용하면 이러한 현상을 일으키고 필요한 산소나 영양분은 혈액 순환에서 공급되어지고 노폐물(탄산가스, 유산)은 제거되어야 하는데 혈액 순환 장애로 인해 산소나 영양분이 충분하지 못하게 되어서 통증을 유발하게 한다.

RSI 증상

가. 손, 팔목, 손가락, 팔, 팔꿈치에는 조이는 감, 불편, 딱딱한 감.

통증이나 화끈거림

나. 손이 저리는 감, 차가운 감, 무감각

다. 손의 힘이나 협동능력 감소나 숨씨가 감소.

라. 저녁에 통증으로 깨어남.

마. 손, 손목, 팔을 스스로 마사지 함.

RSI 증상들도 컴퓨터 증후군에 해당되는 일종의 질병으로 이와 같은 컴퓨터 증후군(VDT 증후군)을 최소화하기 위해서는 실내 조명을 밝게 하고 온도와 습도를 조정하여 쾌적한 환경으로 바꾸어 주고 소음이나 정전기가 발생하지 않도록 해야한다.

모니터는 눈의 위치보다 10~20[cm] 아래에 두고 연속적인 장시간 사용을 지양하고 창가에서 먼 곳에 단말기를 두어 반사를 줄이고 단말기의 화면은 약간 어둡게 하는 것이 눈의 보호에 도움되고 컴퓨터 작업은 하루 4시간 이내로 하는 것이 바람직하며 1시간 이상의 계속적인 작업은 피하고 40~50분 작업 시 10분 가량은 휴식해야 한다.

비타민 C와 칼슘을 섭취하면 체내의 자기파를 해소하고 눈의 VDT증후군을 예방하는데 많은 도움이 된다.

따라서 전자 쑥뜸기 회로의 연구에 있어서는 경, 건, 완(經肩腕)의 가벼운 장애나 예방차원의 기술을 목적으로, 컴퓨터 사용시간을 고려하여 쑥뜸의 시술시간을 40~50분에 시술되게 시간의 조정 한계를 두었다[7].

4. 2 포트블 전자쑥뜸기의 기기 구성

포트블 전자쑥뜸기의 구성은 크게 3가지로 구성되며, 포트블 전자쑥뜸기의 블록 다이어그램은 그림5와 같다.

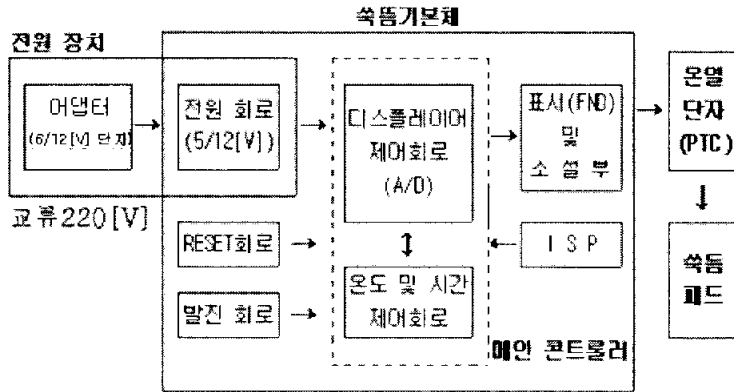


그림 5 포트블 전자쭈뽀기의 블록다이어그램

4. 2. 1 전원부

교류 220[V] 전원을 받아 어댑터를 사용하여 DC 24[V]를 공급하면, 쭈뽀기 본체 내의 정류회로를 거쳐 DC 5/12[V]의 전원을 온열 단자와 제어부에 공급하게 되며 정전압회로를 구성한다.

4. 2. 2 전자 쭈뽀기 기기

메인 컨트롤러는 전원회로, 리셋회로, 발진회로 크게 구성되어지고 발열체 보호회로, 시간 온도 표시부, 전원 스위치 및 시간 온도 조절부, 시간 종료 신호부로 구성되고 금속제 외함으로 외부 충격으로부터 보호된다.

4. 2. 3 온열 단자 및 쭈뽀기패드

본체의 온열부 커넥터에 접속하는 연결 코드와 PTC 세라믹 서미스터 히팅으로 발열하는 단자이고 쭈뽀기 패드 단자와 쉽게 연결되는 구조로 되어있다.

그림6은 온열 단자와 쭈뽀기패드의 구조를 단면으로 나타낸 것이다.

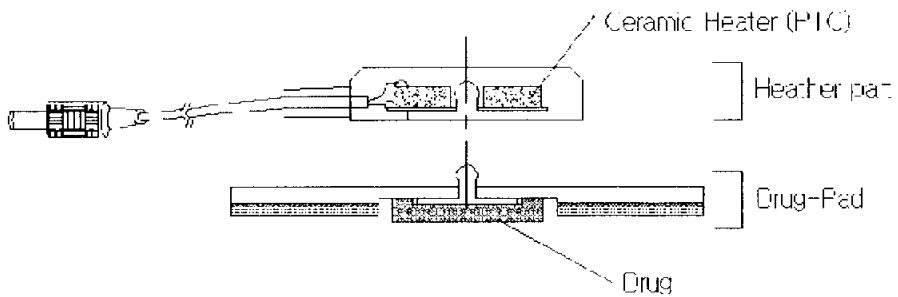


그림 6 온열단자와 쑥뜸패드의 구조도

4. 3 쑥뜸기의 내부 회로 구현

4. 3. 1 메인 콘트롤러

포트를 전자쑥뜸기의 메인 콘트롤러는 8051 계열의 마이크로프로세서로 간단하면서도 기능을 확장한 마이크로프로세서로 미국의 ATMEL사에서 제공하는 CPU인 AT90S4433을 사용하였다. AT90S4433은 AVR<Atmega(Bogen)

Vergad(Wollan) RISC(Reduced Instruction Set Computer)> 마이크로 콘트롤러로 간단히 정리를 하면 다음과 같다[8].

가. 특징

- ㄱ) 고성능 저전력 8-bit RISC(단순 명령어 컴퓨터) 구조의 마이크로 프로세서로 118개의 강력한 명령어가 대부분 1클럭에 1개의 명령어가 실행되고, 8[MHz]에서 8MIPS(Million Instruction Per Second: 초당 명령어 처리 개수)까지의 처리 속도를 가지도록 최적화된 프로세서이다. 또한 32개×8비트의 범용 작업 레지스터를 가지고 있

다.

나) 데이터와 비휘발성 메모리를 갖는다.

4KByte의 ISP(In System Program) 프로그램 플래시(FLASH) 메모리를 내장하고 있어 프로그램을 기록한 후 전원을 제거하더라도 저장된 데이터가 사라지지 않으며, 재 기록이 1000번 이상이 된다. 또한 256 Byte의 ISP EEPROM을 가지기 때문에 제품을 운용하던 중 필요한 데이터 값을 저장해 둘 수 있다.(재 기록100,000번 이상) 그리고 128 Byte의 SRAM을 가지며 LOCK과 EEPROM 데이터 잠금 기능을 가지고 있다.

나) 4[MHz], 3[V], 25[℃]에서의 전력 소모

- ① 활성모드(Active) : 3.4[mA]
- ② 휴식모드(Idle Mode) : 1.4[mA]
- ③ 절전모드(Power-down Mode) : 1[μA] 이하

나. 사양 및 주변 장치

표 2 AT90S4433 마이크로 컨트롤러의 사양 및 주변 장치표

사양 및 주변장치	용량	비 고
FLASH[KB]	4	
EEPROM[Bytes]	258	
RAM[Bytes]	128	
Instruction	118	
I/O Pins	20	포트 B, C, D
Interrupts	14	
Ext. Interrupts	2	
SPI	1	마스터/슬레이브(Master/Slave) 직렬 통신
UART	1	플로그램이 가능한 비동기 직렬 통신
8-bit Timer	1	별도의 프리스케일러를 갖은 타이머/카운터
16-bit Timer	1	별도의 프리스케일러, 비교, 캡처, 8,9,10비트 PWM 기능을 갖은 타이머/카운터
PWM	2	Pulse Width Modulation
Watchdog Timer	Y	별도의 내부 오실레이터를 가짐 플로그램이 가능
Analog Comp	Y	내부 아날로그 비교기
10-bit A/D Channels	6	6채널 10비트 아날로그디지털변환기(ADC)
On-Chip Oscillator	·	
Brown Out Detector	Y	
ISP	Y	In System Programming
Vcc[V]	4-6	본체에서는 5 [V] 사용
Clock speed[Mhz]	0-8	본체에서는 4 [Mhz] 주파수 사용
Packages		본체에서는 PDIP형(28핀) 사용

다. 핀의 기능

AT90S4433의 PDIP(28핀)의 핀 배치는 그림10과 같다.

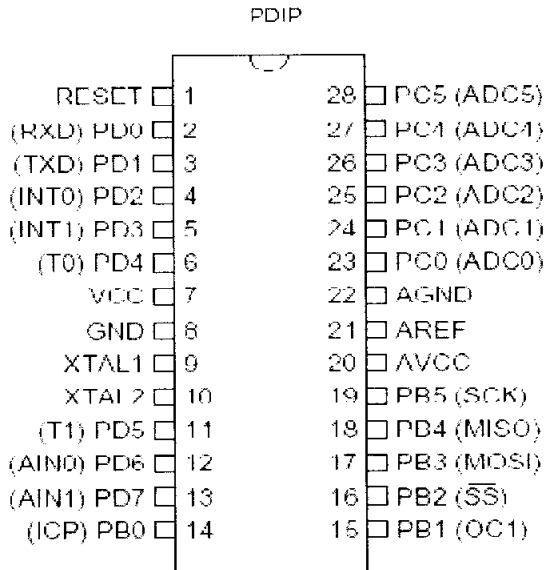


그림 7 AT90S4433 핀 구성도

가) 입출력 핀(I/O Pins)

- ① Port B : 6비트 입출력 양방향성 포트로서 내부 Pull-up 을 동작시킬 수 있고, 출력 버퍼는 20[mA]의 전류까지 싱크(sink) 할 수 있다.

표 3 AT90S4433 마이크로 컨트롤러의 I/O 핀(Port B)의 기능표

핀번호	포트핀	특 수 기 능
14	PB0	ICP(입력캡처 노이즈 제거, Bit 0)
15	PB1	OC1(시간 함수를 위한 PWM모드 출력, Bit 1)
16	PB2	SS (슬레이브 포트 선택 입력)
17	PB3	MOSI(SPI 마스터 데이터 출력/슬레이브 데이터 입력)
18	PB4	MISO(SPI 마스터 데이터 입력/슬레이브 데이터 출력)
19	PB5	SCK(SPI 마스터 클럭 출력/슬레이브 클럭 입력)

* 참고 ICP(Input Capture Pin), OC1(Output Compare Match output),

MOSI(Master Out Slave In), MISO(Master In Slave Out), SCK(SPI CLOCK)

- ② Port C : 6바이트 입출력 양방향성 포트로서 내부 Pull-up을 동작시킬 수 있고, 출력 버퍼는 20[mA]의 전류까지 싱크(sink)할 수 있다. 그리고 A/D변환기에서 아날로그 입력을 받는다.
핀 번호(23~28), 포트 핀(PC0~PC5)
- ③ Port D : 8바이트 입출력 양방향성 포트로서 내부 Pull-up을 동작시킬 수 있고, 출력 버퍼는 20[mA]의 전류까지 싱크(sink)할 수 있다.

표 4 AT90S4433 마이크로 컨트롤러의 I/O 핀(Port D)의 기능표

핀번호	포트 핀	특 수 기 능
2	PD0	RXD(데이터 수신 : UART에 대한 데이터 입력 핀)
3	PD1	TXD(데이터 송신 : UART에 대한 데이터 출력 핀)
4	PD2	INT0(외부인터럽트0)
5	PD3	INT1(외부인터럽트1)
6	PD4	T0(타이머/카운터0 입력 핀)
11	PD5	T1(타이머/카운터1 입력 핀)
12	PD6	AIN0(아날로그 비교기 0 입력)
13	PD7	AIN1(아날로그 비교기 1 입력)

나) 기타 기능 핀

표 5 AT90S4433 마이크로 컨트롤러의 기타 기능 핀의 기능표

핀번호		기 능
7	VCC	마이크로 프로세서가 동작하도록 공급하는 전원(5[V])
8	GND	신호의 접지 핀으로 사용
1	<i>RESET</i>	리셋입력으로 50[ns]이상의 "LOW Level"입력이 지속되면 리셋이 발생한다.
9	XTAL1	수정발진기와 내부 클럭 동작회로의 입력단자
10	XTAL2	수정발진기의 출력단자
20	AVCC	포트 A와 A/D 변환기의 전원공급전압
21	AREF	A/D 변환기의 아날로그 기준 입력전압
22	AGND	

라. AT90S4433의 블록다이어그램

Figure 1. The AT90S4433 Block Diagram

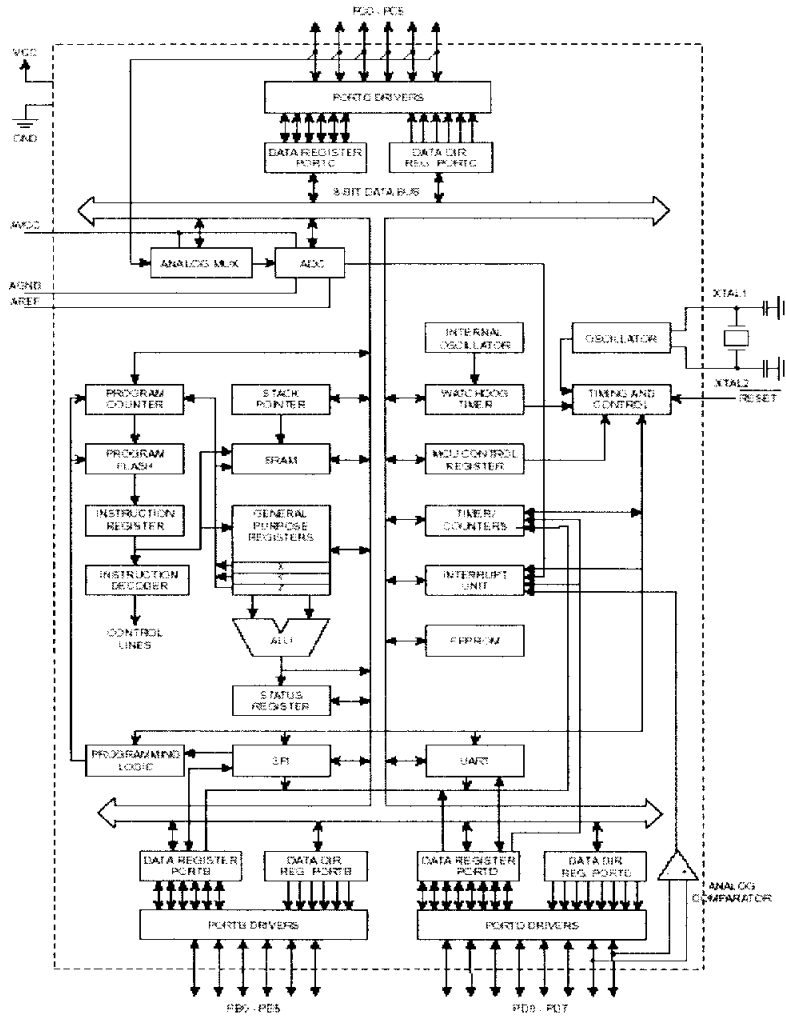


그림 8 AT90S4433의 블록다이어그램

4. 3. 2 디스플레이(FND) 회로

LED 디스플레이는 SND427D를 2개 사용하여 온도와 시간을 각기 표시하도록 하였고, 일반적인 FND 제어회로에는 7447 IC를 이용하는 데 비하여 본 실험 회로에서는 CPU의 8비트를 연결하여 직접 제어하도록 하였다. FND를 구동하기 위한 메인 컨트롤러는 Port D(I/O핀 8개)와 Port C(PC2~PC5, 4개 : ADC)를 사용하였다. 각 세그먼트 LED에는 전류 제한용 저항을 외부에 직렬로 연결하여 사용한다.

4. 3. 3 전원회로

전원 회로는 입력전원에 상관없이 항상 DC전압을 출력 할 수 있도록 설계되었다. 입력전원에 교류전압이 들어오면 어댑터에서 트랜스를 통과한 전압이 브리지형다이오드 정류회로를 거치면 항상 일정한 극성을 가지는 직류 전원으로 온열부와 내부회로에 전원을 공급하게 된다.

또한 직류 전원이 공급될 때 공급전원의 극성이 바뀌어도 출력 측의 극성은 항상 일정한 극성을 갖게 되도록 브리지회로가 구성되어 있다. VCC로 내부전원을 공급하는 단자는 2개의 저항이 연결되어 있으며, 공급전원을 분할하는 기능으로 어댑터를 통과한 12[V] 전원은 PCT 발열부 전원으로, 분할 저항에 의해 분할된 전원은 마이크로프로세스 및 내부회로의 전원으로 사용할 수 있도록 하였다.

4. 3. 4 발진회로

마이크로프로세서를 일정 간격으로 동작시키는 데 매우 중요한 부분으로 4[MHz]수정 발진기를 사용하였다. 클럭 주기는 수정 발진기의 정격과 관계가 있으므로 1 클럭 주기는 $\frac{1}{4} \times 10^{-6}[\text{sec}]$, 약 $0.25[\mu\text{s}]$ 이다.

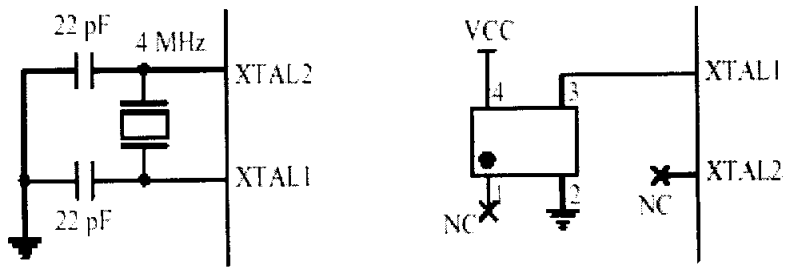


그림 36 크리스털 오실레이터의 접속도

4. 3. 5 리셋회로

마이크로 컨트롤러의 상태를 초기 상태로 만들어 주는 것으로 다음과 같은 3가지의 리셋 구조를 가지고 있다.

1) 전원 리셋(Power-On Reset)

Vcc와 GND에 전원이 공급될 때 MPU(Microprocessor Unit)에 리셋이 걸리게 되는 Reset이다. 전원 리셋 회로는 Vcc가 안정한 영역에 도달할 때까지 소자가 작동하지 않게 한다.

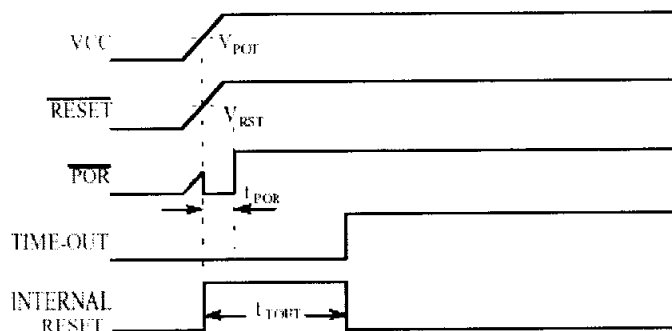


그림 10 MCU 스타트업 빠른 상승 VCC와 RESET 연결

2) 외부 리셋(External Reset)

리셋 핀에 전압이 High에서 Low로 천이 될 때 리셋이 걸리게 된다. 단, Low 레벨의 신호가 XTAL 클럭의 2회 이상의 시간을 가져야 한다.

리셋 핀은 칩 내부에 풀업 되어 있기 때문에 외부 리셋이 필요 없을 경우는 연결하지 않아도 된다. 따라서 리셋은 Vcc단자와 연결한 것과 같은 효과를 나타낸다. 전원 공급 후 리셋 핀에 Low 레벨의 신호를 인가 한 후 이것을 지속함으로써 전원 리셋 시간을 연장할 수 있다.

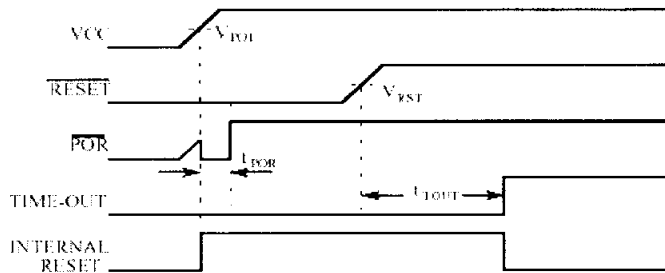


그림 11 MCU 스타트업 외부적으로 RESET 제어

3) 워치독 리셋 (Watchdog Reset)

워치독 설정 시간이 되면 크리스털 2클럭 주기의 짧은 리셋 펄스가 발생한다. 이 펄스의 하강 에지 생성 때 동일하게 시간 지연이 일어난다.

제 5장. 내부 프로그래밍(Software)

AVR은 내부의 플래시 메모리와 EEPROM에 데이터를 써넣을 수 있도록 SPI 인터페이스를 제공한다. SPI는 오직 3라인을 이용한 통신 방법으로 MOSI(Master Out Slave In), MISO(Master In Slave Out), SCLK(SPI CLOCK) 신호를 이용한다. Motorola에서 개발되었으며 Master와 Slave가 SCLK에 동기하여 데이터를 교환하는 방식이다. 플래시 메모리를 액세스 하기 위해서는 AVR의 Reset 핀을 low로 한 상태에서 앞의 3가지 신호를 이용하여 데이터를 읽고 쓰기가 가능하다. 즉, AVR과의 SPI 인터페이스를 맞추어 주기만 하면 내부 플래시 메모리의 액세스가 가능하다는 것이다. 이는 AVR의 프로그래머를 싼값으로 구현하게 하는 동기가 된다.

AVR 프로그래머의 구현은 ISP형태와 롬라이터 형태로 구현할 수 있다.

5. 1 메모리 구조

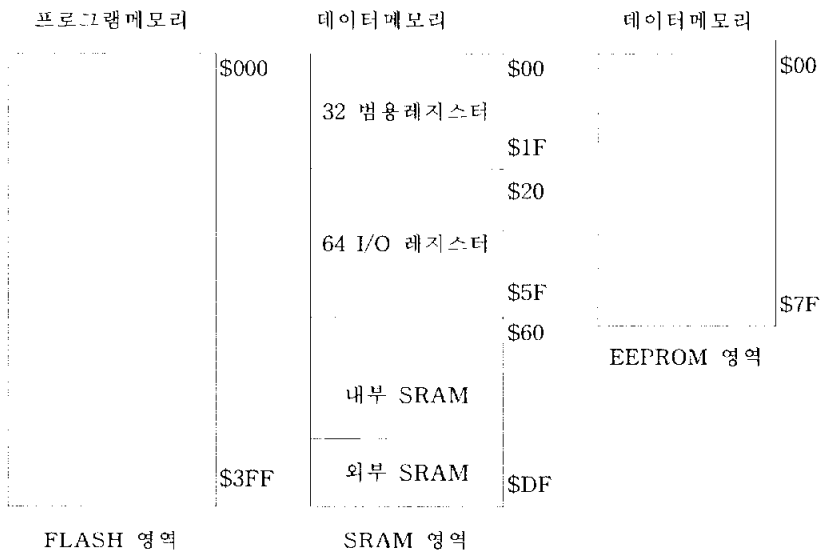


그림 12 메모리의 구조

- 프로그램 메모리 공간 : 0x00 ~ 0x1F(1KWords)
- 범용 레지스터 공간 : 0x00 ~ 0x1F(32Byte)
- 입출력 레지스터 : 0x20 ~ 0x5F(64Byte)
- SRAM 영역 : 0x60 ~ 0xDF(128Byte)

5. 1. 1 프로그램 메모리(FLASH 메모리)

프로그램 메모리는 플래시 메모리라고도 하며 AVR 종류에 따라 크기가 결정되는데, 소스프로그램을 컴파일 하여 마이크로프로세서가 인식하는 HEX파일을 저장하는 장소이다.

5. 1. 2 EEPROM 메모리

비휘발성으로 전기적 신호에 의해 데이터를 기록하거나 지우는 메모리로서 프로그램일 수정하거나 다시 입력할 때 기존의 ROM을 그대로 이용 할 수 있다는 장점이 있다.

5. 1. 3 SRAM 메모리

SRAM은 데이터 메모리로서 32개의 범용레지스터와 64개의 I/O 레지스터, 내부 SRAM, 외부 SRAM 영역으로 이루어진다. 32개의 범용레지스터는 내부 메모리로 프로세스가 프로그램을 실행하는 도중 연산 결과나 데이터를 임시로 저장해 두는 영역이며, 간접 어드레스 지정 포인터 레지스터로 사용되는 특수레지스터를 가지며, ALU는 상수와 레지스터 간 또는 레지스터와 레지스터간의 대수 및 논리 함수의 연산을 지원한다. 한 개의 레지스터 명령은 ALU에서 처리되며 레지스터 명령을 추가하여 일반적인 메모리 어드레스 지정 모드도 역시 레지스터 파일로 사용될 수 있다.

5. 2 ISP(In System Programmer)

시스템에 프로그래밍을 위한 회로와 커넥터가 마련이 되어 있어서 따로 롬라이터를 이용하지 않고도 PCB 기판 상에서 프로그래밍이 가능하도록 하며, 일반 롬라이터는 전원과 클럭을 롬라이터 자체적으로 공급해야 하는 반면에 ISP는 PCB 회로 상의 전원과 클럭을 이용한다. 최근에 인기 있는 방식으로 약간의 회로 추가로 시스템의 디버깅이 용이하고 업그레이드가 쉬운 시스템을 만들 수 있다는 장점이 있으나 부피가 커지고 제품 단가가 올라간다는 단점이 있다.

5. 3 사용 소프트웨어

Kanda사에서 공급받아서 ATMEL에서 판매하는 Starter Kit로 STK 200을 사용하였다. STK는 기본적인 테스트를 할 수 있는 evaluation board와 ISP 프로그래머를 포함하고 있다.

ISP 프로그래머로는 AVR-ISP 소프트웨어(V2.65)를 사용한다[9].

5. 4 전원부 및 콘트롤 회로 구성도

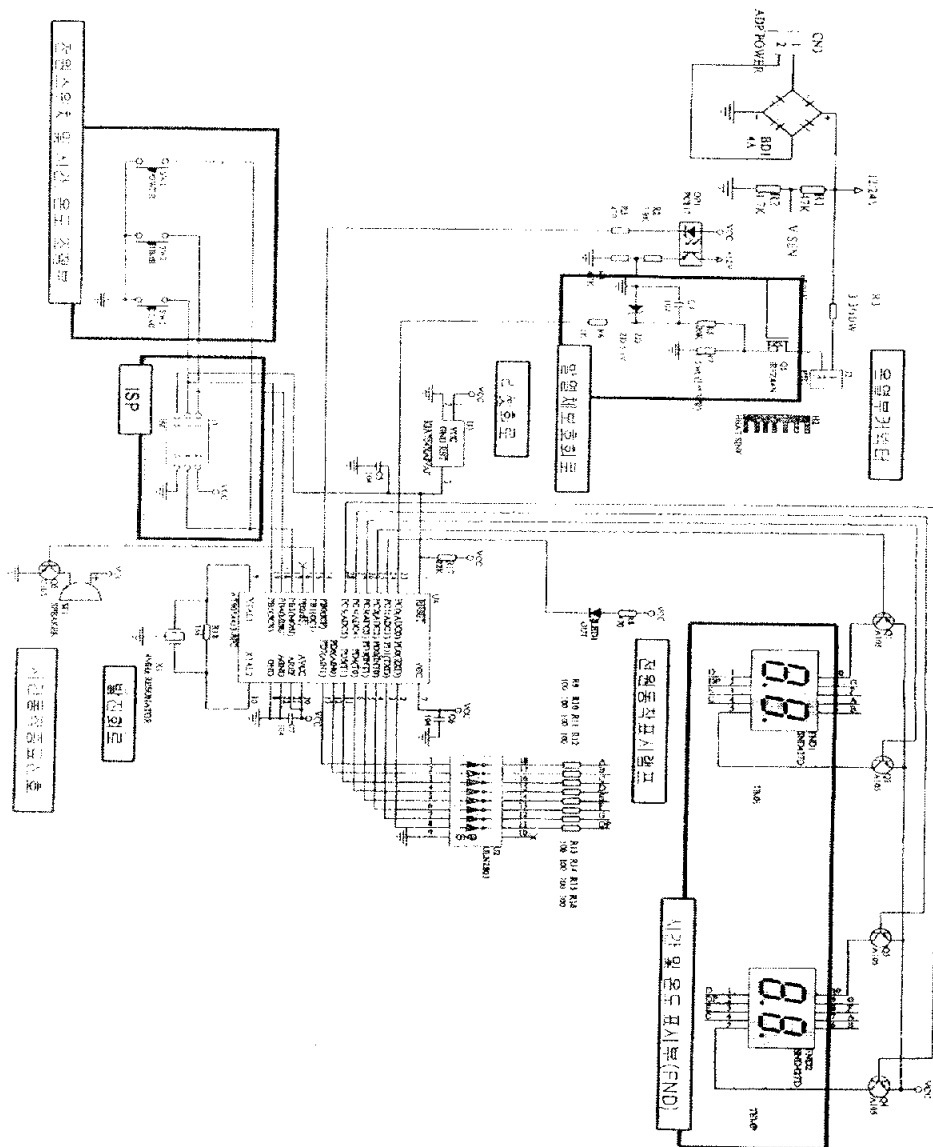


그림 13 포터블 전자속도계의 회로도

5. 5 회로의 기본 설정 및 신호 동작

AC 220[V], 직류전원 12[V] 어댑터를 전자 쏙뚝기에 전원으로 주면 12[V]측 전원은 온열단자에 보내져 쏙뚝이 사출되게 하고 분할저항을 거친 전압은 5[V]로 만들어져 마이크로세서와 전자 신호회로의 Vcc 전압으로 사용된다.

5. 5. 1 전원 S/W ON4 조작과 설정

- 가. 전원 전원 S/W를 누르면 입력 LAMP에 표시등이 점등된다.
- 나. 제어부 (마이크로 컨트롤러) 프로그램에 의해 초기화 값이 FND에 표시된다.
- 다. 초기화 값은 시간 설정 40분, 온도설정 43[°C]으로 정한다.
- 라. 온도 조절 버튼과 시간설정 버튼으로 설정온도, 시간을 정한다.
온도는 체온보다 높은 38[°C]를 최저로 하고 49[°C]까지를 내부 설정하여 버튼 조작시 2[°C]씩을 가감 할 수 있다.
- 마. 시간 조절은 0~40 분으로 설정하고 버튼 조작으로 10분 간격으로 변환되고 조정하도록 프로그램화하였다. 종료 시에 스피커음으로 종료 신호를 발생시킨다.

5. 5. 2 설정온도의 유지

- 가. 제어신호가 설정된 값을 디스플레이(FND)에 온도와 시간 값으로 출력되고 포토 커플러(Photo Coupler) Pc 817의 전압에 의해 N형 MOSFET (IRFZ44N 사용)가 동작하게 된다. MOSFET는 게이트(G)전압이 0 일 때 D-S전류 (드레인-소스전류)가 흐르지 않게 되고 반대의 12[V]전압이 걸릴 때에 많은 양의 전류가 흐르게 된다. 그러므로 제어부의 조절 신호 값에 의해서 MOSFET의 전류 값이 제어되므로 일정온도가 유지된다.
- 나. 온열단자는 조절 온도 보다 높아지면 저항 값이 증가되어 PTC

세라믹 발열체의 전류 값이 떨어지게 되고 그 전류값은 마이크로 콘트 롤러의 프로그래밍된 신호값(5v) 에 의해서 작동(12v) 되어 MOSFE-T를 조절하여 D-S사이의 전류값이 증가되어 입력된 일정한 온도로 유지한다.

5. 5. 3 온열부

온열단자의 커넥터를 쑥뜸기 본체에 연결하면 두단자에 제어된 직류전원(전압 12[V]전류 1[A])이 인가되어 온열단자의 PTC 발열체가 발열하여 원적외선을 피부에 전달하면서 쑥뜸패드의 쑥뜸로선을 녹여서 피부에 스며들게 하고, 녹여 내린 약쑥 성분과 온열이 혈액순환을 좋게하고 정맥 부위의 파하에 깊숙이 침투시켜 쑥뜸을 시술되게 한다.

그림 14는 포트블 전자쑥뜸기의 완제품 사진으로서 쑥뜸기의 본체 어댑터, 온열단자, 쑥뜸로선, 쑥뜸패드(온열단자에 연결)를 동시에 보여주고 있다.

그림 15는 포트블 전자쑥뜸기 내부의 회로기판을 제작된 상태로 보여주고 있다.

포트블 전자썩뜸기의 완성품은 그림 14, 15와 같다.

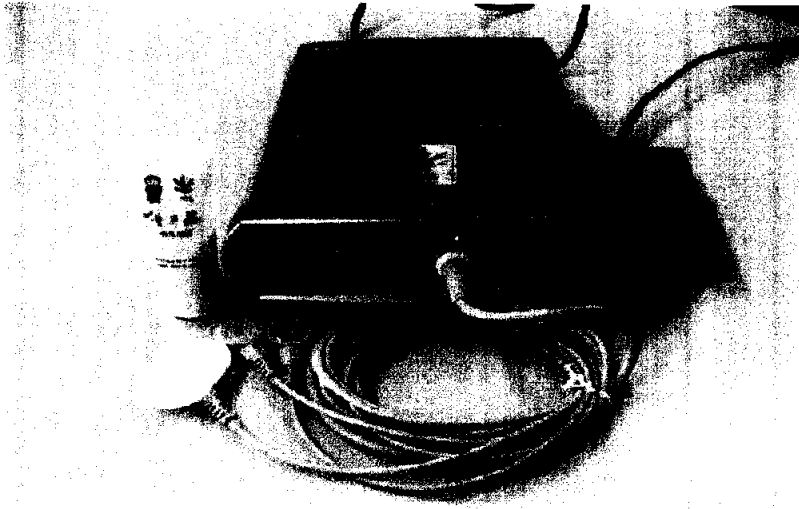


그림 14 포트블 썩뜸기 사진

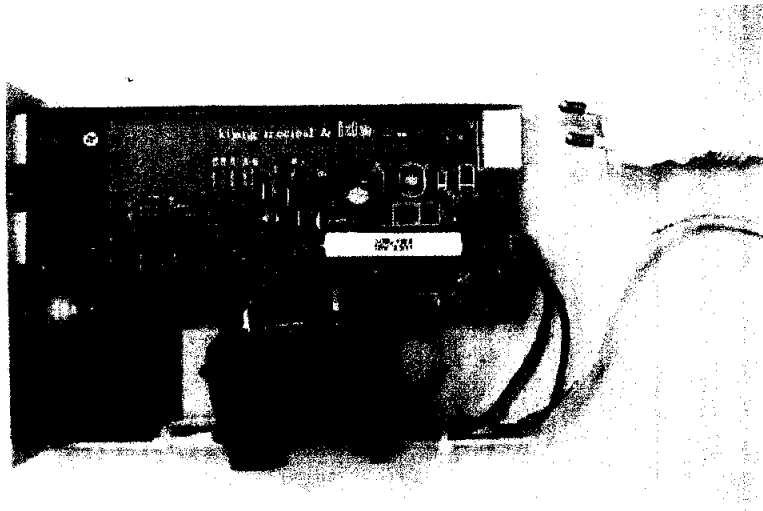


그림 15 포트블 썩뜸기의 내부 사진

제 6장 결론

PC로 업무를 하는 분야는 이제 일반적인 사무는 물론이고 전문적인 업무와 생산공정과 일반가정에서도 널리 활용되고 있다. 컴퓨터는 업무와 생활을 혁신하였지만 컴퓨터 작업자는 연속되고 과중한 업무로, 경관완(經肩腕)의 고통과 장애를 겪고 있다.

기존의 컴퓨터 내장형 전자쑥뜸기는 시술자가 컴퓨터 앞에서 40분가량을 이동하지 못하고 시술하므로 사부이동시에 시술을 중단하는 불편이 있었으나 본 연구의 포트블 전자 쑥뜸기는 어느 정도는 이동하여도 시술이 가능하게 되었다. 이는 쑥뜸기 1대로서 위치를 옮겨가며 여러 사람이 번갈아 가면서도 시술을 할 수 있도록 구현된 것이다.

컴퓨터 사용자가 VDT증후군의 치료 및 예방을 위해서 쑥뜸을 하는 것은 온열치료와 약쑥의 약리 성분을 피하에 전달시켜 혈액순환을 원활하게 하고 산소와 기타 영양분을 말초신경에 전달하여 노폐물 배출과 건강한 상태의 신체를 유지함으로서 삶의 질과 업무의 효율화를 기하기 위함이다.

쑥뜸은 동양의학 중 가장 중요한 치료의 한가지 방법이다. 그러나 전통적인 쑥뜸 시술 시에는 여러 가지 불편이 있었고 비위생적, 비 과학화로 점차 외면되어왔으나 전자쑥뜸기의 개발로서 다양한 전원을 이용하게 되고 시술 부위도 종전의 쑥뜸에서 시술하지 못하던 부위까지 시술되도록 개발되어 ‘우리 것을 세계화’ 하도록 과학적인 연구가 계속되고 있다.

전자쑥뜸기의 시술 효과는 여러 연구자료에 나와 있듯이 그 효능이 임상실험에서 입증되고 있다. 전자쑥뜸기가 컴퓨터 사용자들에 의해 많이 시술되고 그 효능들이 건강유지에 도움이 되고 가벼운 질병 치료에 유용하다는 것이 알려진다면 건강하고 밝은 사회에 일조 할 것이다. 그리고

앞으로의 연구에서는 AC 220V를 사용하지 않고 DC(건전지나 배터리)를 전원으로 활용하는 전자 쑥뜸기 등으로 연구 발전하여 특정 장소 등 전기가 없는 지역(혹한지), 버스, 기차, 선박여행, 등산, 낚시 중에도 시술할 수 있는 기술 등이 연구되어 휴대용도 보급 할 수 있을 것이다.

참 고 문 헌

- [1] Jong Il Bae et al, “Development of the Heating Pad with the Artemisia Patch for Moxibustion Remedy,” The 5th Asia-Pacific Conference on Medical and Biological Engineering, 2000
- [2] Jong Il Bae et al, “The Experiment Estimation of the Effect on the Body Heat by the Moxa Caulerizer and the Artemisia Lotion,” The 5th Asia-Pacific Conference on Medical and Biological Engineering, 2000
- [3] Jong Il Bae et al, “The Implement of the Drug-Pad Moxibustio Caulerizer with The Single Heating Pad & The Multiple Heating-Pad,” The 6th Asia-Pacific Conference on Medical and Biological Engineering, 2001
- [4] 이성훈, 쑥뜸관을 갖는 새로운 간접구 방식의 구현, 부경대학교석사학위논문, 2000
- [5] 민강옥 지, 온열 및 수치료 도서출판 대학서림, 1993
- [6] URL:<http://user.chyollian.net/~pain7575/neck5.htm>
- [7] URL:<http://user.chyollian.net/~pain7575/vdt.htm>
- [8] URL:<http://www.atmel.com>, AT90S4433의 매뉴얼, ATMEL사
- [9] URL:http://micro.new21.org/avr/html/avr_prog.htm
- [10] 양주완, PTC 발열재를 이용한 전자쑥뜸기 구현, 부경대학교석사학위논문, 2000
- [11] 윤동엽, 쑥엑스제와 전자쑥뜸기를 사용한 피부온도 상승반응의 고찰, 부경대학교석사학위논문, 2000
- [12] 공기홍, 차광용 전자쑥뜸기의 구현, 부경대학교 석사학위논문, 2001
- [13] 김은수, 전자쑥뜸기 다중 발열패드의 구현, 부경대학교석사학위논문, 2001
- [14] 서정필, 머리 쑥뜸용 장치의 개발에 관한 연구, 부경대학교석사학위논문, 2001
- [15] 기타 관련사이트 인터넷(InterNet) 정보 검색

- [16] 이만섭, 차량용 전자 쉑튐기의 회로 구성 및 동작 특성에 관한 연구, 부경대학교 석사 학위 논문, 2003
- [17] 김병석, Personal Computer용 전자 쉑튐기 회로의 구현, 부경대학교 석사 학위논문, 2003

부 록

////////////////////////////////////

/*

PC/ADP MEDICAL

TARGET - AT90S4433
CLOCK - 4MHz RESONATOR
COMPILER - IAR

////////////////////////////////////

#include <io4433.h>
#include <macros.h>
#include <ina90.h>
#include <pc.h>
#include <beep.h>

flash char FND_FONT[] = { 0x88, 0xBE, 0xC4, 0xA4, 0xB2, 0xA1, 0x81,
0xBC, 0x80, 0xA0 };
char FND_RAM[4];
char fnd = 0;
char fndblank;
char FND;

flash char TEMP_FONT[] = { 0, 38, 41, 43, 45, 47, 49 };
flash char HUGE_TABLE[] = { 0, 15, 10, 7, 4, 2, 1 }; //sec
//flash int TEMP_TABLE[] = { 0, 200, 150, 102, 74, 63, 45 };
flash int TEMP_TABLE[] = { 0, 220, 200, 180, 150, 100, 80 };
//#define SHORT_VALUE 430 //2.1A
#define MIN_VOLTAGE 164 //8V
#define LEAK_VALUE 30 //150mA

```

#define NONE_PTC      20 //100mA
#define TOLERANCE     22 //diode drop voltage
#define OVER_PTC      225 //1.1A at 1min

#define PER 18
long VOLTAGE;
int SHORT_CURRENT, SHORT_VOLTAGE, RESISTOR = 57; //5.7ohm
int VOLT_CNT; //voltage drop check time, 30sec at 12V, 1sec at 32V

#if 0
        **** ERROR DEFINE ****
        ERROR 1 - PTC OVERLOAD
        ERROR 3 - PTC LEAK
        ERROR 6 - LOW VOLTAGE ERROR
        ERROR 8 - PTC CONNECT ERROR
        ERROR 10 - PTC OVER
        //////////////////////////////////////
#endif

#define ERROR_CNT 20
#define SHORT_CNT 2

#define MAX_AD_CHANNEL 2
char ad_chan = 0; //A/D channel
char ad_enable = 1;
char ad_end = 0; //AD 변환완료
int CHAN_AD, VOLT_AD;
#define CHAN 0
#define VOLT 1

//parametmer
char optime; //operating time(0 ~ 40)
char oplevel; //operating level(1 ~ 6)

```

```

unsigned int delaytimer;
extern int beep_time;

char blank_cnt;
char powerflag = 0;
char stopflag = 0;
char errorflag = 0;
char poweroffflag = 0;

char errorbeep = 0;
char endbeep = 0;
char errordisp = 0;

char ptestatus = 0; //0 = off, 1 = on
char ptchugetime;
char ptcnt;
char detectenable;

char overloadcnt;
char leakcnt;
char connectcnt;
char lowvoltcnt;
int dropvoltcnt;

char offtime = 0;
char overtime;

#define KEY_DELAY 3
char powerkey, levelkey, timekey;
char powerkeycnt, levelkeycnt, timekeycnt;

void sleep(unsigned int t)
{
    while(--t) _WDR();
}

```

```

}

void delay(unsigned int t)
{
    delaytimer = t;
    while(delaytimer) _WDR();
}

char msectimer;
int sectimer;
char mintimer;
interrupt [TIMER0_OVF_vect] void main_timer(void)
{
    static char tmr0cnt = 0;

    TCNT0 = -4; //256usec * 4 = 1.024msec

    if (++tmr0cnt >= 4)
    {
        tmr0cnt = 0;

        PORTC |= 0x3C;
        _NOP();
        FND &= 0x80;
        FND |= (FND_RAM[fnd]&0x7F);
        PORTD = FND;

        if (fnd == 0)
        {
            PORTC &= 0x1C;
            ++fnd;
        }
        else if (fnd == 1)

```

```

    {
        PORTC &= 0x3B;
        ++fnd;
    }
    else if (fnd == 2)
    {
        PORTC &= 0x2F;
        ++fnd;
    }
else
    {
        //key read
        if (!(PINB & 0x08))
        {
            if (++powerkeycnt >= KEY_DELAY)
            {
                powerkeycnt = 0;
                powerkey = 1;
            }
        }
        else
        {
            powerkey = 0;
            powerkeycnt = 0;
        }

        if (!(PINB & 0x10))
        {
            if (++levelkeycnt >= KEY_DELAY)
            {
                levelkeycnt = 0;
                levelkey = 1;
            }
        }
    }

```

```

        else
        {
            levelkey = 0;
            levelkeycnt = 0;
        }

    if (!(PINB & 0x20))
    {
        if (++timekeycnt > KEY_DELAY)
        {
            timekeycnt = 0;
            timekey = 1;
        }
    }
    else
    {
        timekey = 0;
        timekeycnt = 0;
    }

    PORTC &= 0x37;
    fnd = 0;
}

}

if (++msectimer >= 10) //10msec
{
    msectimer = 0;

    if (delaytimer) --delaytimer;
    if (beep_time) --beep_time;

    if (optime)
    {

```

```

//ad conversion
    if (ad_enable)
        adc_start();

    if (VOLT_AD < MIN_VOLTAGE) //low voltage
error
    {
        if (++lowvoltcnt >= ERROR_CNT)
            errorflag = 6;
    }
    else if (VOLT_AD < SHORT_VOLTAGE) //voltage
drop error
    {
        if (++dropvoltcnt >= VOLT_CNT)
            errorflag = 1;
    }
    else
    {
        lowvoltcnt = 0;
        dropvoltcnt = 0;
    }

    if (ptcstatus == 1 && !detectenable) //ptc on status
    {
        if (!ptccnt)
        {
            if (CHAN_AD >= NONE_PTC) //PTC가 동작중 삽
                errorflag = 8;
        }
        else
        {
            if (CHAN_AD < NONE_PTC) //PTC가 동작중 빠진
경우

```

```

        errorflag |= 8;
    else if (CHAN_AD <= TEMP_TABLE[oplevel])

//stop
    {
        PORTB &= 0xFE; //PTC off
        FND &= 0x7F;
        PORTD = FND;
        ptcstatus = 0; //not operating
        overloadcnt = 0;
        detectenable = 1;
    }
}

if (CHAN_AD >= SIHORT_CURRENT) //overload error
{
    if (++overloadcnt >= SIHORT_CNT)
        errorflag |= 1;
}

ptchugetime = HUGE_TABLE[oplevel];
}
else if (ptcstatus == 0 && !detectenable) //ptc1 off status
{
    if (!ptchugetime)
    {
        PORTB |= 0x01; //PTC on
        FND |= 0x80;
        PORTD = FND;
        ptchugetime = HUGE_TABLE[oplevel];
        ptcstatus = 1; //operating
        detectenable = 1;
    }
}
else
{

```

```

        //--ptchugetime[0];
    if (CHAN_AD >= LEAK_VALUE) //leak error
    {
        if (++leakcnt >= ERROR_CNT)
            errorflag = 3;
    }
    else
        leakcnt = 0;
}
}
}

if (powerflag == 1)
{
    if (++sectimer >= 976) //1 sec
    {
        sectimer = 0;

        if (!optime) //timer blank
        {
            if (!fndblank)
            {
                fndblank = 1;

                if (!errorflag)
                {
                    time_display(optime);
                    level_display(oplevel);

                    endbeep = 1;
                }
            }
            else
            {

```

```

        if (errordisp)
            time_display(errorflag);
        else
            time_display(BAR);
        level_display(ERR);

        errorbeep = 1;
    }
}
else
{
    fndblank = 0;

    time_display(OFF);
    level_display(OFF);
}

if (++blank_cnt >= 60) //1분 후 자동 종료
    poweroffflag = 1;
    //power_off();
    //powerflag = 2;
}
else //operating
{
    if (ptchugetime) --ptchugetime;
    if (detectenable) --detectenable;

    if (overtime)
    {
        --overtime;
        if (!overtime) //ptc over detect
        {
            if (CHAN_AD >= OVER_PTC) //PTC가 4

```

개 이상인 경우

```

errorflag = 10;
    }
}

if (++mintimer >= 60) //1 min
{
    mintimer = 0;
    --optime;
    time_display(optime);

    if (!optime)
    {
        stopflag = 1;
        fndblank = 1;
    }
}

if (offtime) --offtime;
}

}

interrupt [ADC_vect] void ADC_interrupt(void)
{
    static long ad[MAX_AD_CHANNEL];
    static char cnt[MAX_AD_CHANNEL];

    if (ad_chan == CHAN)
    {
        ad[CHAN] += ADC;

        if (++cnt[CHAN] >= 3)
        {

```

```

        CHAN_AD = ad[CHAN]/cnt[CHAN];
        cnt[CHAN] = 0;
        ad[CHAN] = 0;
    }
}
else if (ad_chan == VOLT)
{
    ad[VOLT] += ADC;

    if (++cnt[VOLT] >= 3)
    {
        VOLT_AD = (ad[VOLT]/cnt[VOLT]) *
TOLERANCE;
        cnt[VOLT] = 0;
        ad[VOLT] = 0;
    }
}

if (++ad_chan >= MAX_AD_CHANNEL)
    ad_chan = 0;

ad_end = 0;
}

void adc_enable(void)
{
    ad_enable = 1;
}

void adc_disable(void)
{
    ad_enable = 0;
}

```

```

void adc_start(void)
{
    if (!ad_end)
    {
        ADMUX = ad_chan;
        ADCSR |= 0xC8;

        ad_end = 1;
    }
}

void initialize(void)
{
    PORTB = 0x3A;
    PORTC = 0x1C;
    PORTD = 0;

    DDRB = 0x07;
    DDRC = 0x3C;
    DDRD = 0xFF;

    ACSR = 0x80; //comparator disable

    ADMUX = 0x00; //ADC0 select
    ADCSR = 0x85; //ADC enable, CK/32

    TCCR0 = 0x05; //CK/1024 = 256usec
    TCNT0 = -4; //256usec * 4 = 1.024msec
    TIMSK = 0x02; //TMR0 overflow enable

    //MCUCR = 0x20; //sleep enable, LOW level interrupt

    FND = 0x7F;
    powerkey = 0;

```

```

    levelkey = 0;
    timekey = 0;

    WDTCR = 0x0E; //watchdog enable
}

void time_display(char time)
{
    if (time == OFF)
    {
        FND_RAM[0] = 0xFF;
        FND_RAM[1] = 0xFF;
    }
    else if (time == BAR)
    {
        FND_RAM[0] = 0xF7;
        FND_RAM[1] = 0xF7;
    }
    else
    {
        if (time < 10)
            FND_RAM[0] = 0xFF;
        else
            FND_RAM[0] = FND_FONT[time/10];
        FND_RAM[1] = FND_FONT[time%10];
    }
}

void level_display(char level)
{
    if (level == OFF)
    {
        FND_RAM[2] = 0xFF;
        FND_RAM[3] = 0xFF;
    }
}

```

```

    }
    else if (level == BAR)
    {
        FND_RAM[2] = 0xF7;
        FND_RAM[3] = 0xF7;
    }
    else if (level == ERR)
    {
        FND_RAM[2] = 0xC1;
        FND_RAM[3] = FND_FONT[1];
    }
    else
    {
        if (!level)
            FND_RAM[2] = 0xFF;
        else
            FND_RAM[2] = FND_FONT[TEMP_FONT[level]/10];
        FND_RAM[3] = FND_FONT[TEMP_FONT[level]%10];
    }
}

void power_on(void)
{
    power_on_beep();
    powerflag = 1;
    optime = 40;
    oplevel = 3;

    start();
}

void start(void)
{
    PORTB &= 0xFE; //PTC off

```

```

//PORTC &= 0x3F;

time_display(optime);
level_display(oplevel);

FND |= 0x80;
PORTD = FND;
delay(30);
FND &= 0x7F;
PORTD = FND;
delay(30);
FND |= 0x80;
PORTD = FND;
delay(30);
FND &= 0x7F;
PORTD = FND;
//delay(30);

msectimer = 0;
sectimer = 0;
mintimer = 0;
blank_cnt = 0;
ptchugetime = 5 * offtime;
overtime = 60 + ptchugetime;
ptcstatus = 0;
errordisp = 0;

errorflag = 0;
overloadcnt = 0;
leakcnt = 0;
connectcnt = 0;
lowvoltcnt = 0;
dropvoltcnt = 0;

```

```

        ptc_search();
        //start_beep();
    }

void ptc_search(void)
{
    int adv;

    detectenable = 1;

    //voltage test
    if (VOLT_AD < MIN_VOLTAGE) //low voltage error
    {
        errorflag = 6;
        return;
    }
    else
    {
        VOLTAGE = (long)VOLT_AD*100;

        SHORT_CURRENT = VOLTAGE/RESISTOR;
        SHORT_CURRENT = SHORT_CURRENT -
((SHORT_CURRENT*PER)/100); //percentage of max current

        VOLTAGE /= 100;
        VOLTAGE *= 488;
        VOLTAGE /= 1000;
        VOLT_CNT = ((320-VOLTAGE)*350)/20; //drop check time

        SHORT_VOLTAGE = ((VOLTAGE*77)*10)/488; //77% drop
check
    }

    //ptc search

```

```

    PORTB |= 0x01; //PTC on
    //FND |= 0x80;
//PORTD = FND;
    delay(30);
    adv = CHAN_AD;
    if (adv > SHORT_CURRENT) //overload
    {
        PORTB &= 0xFE; //PTC off
        errorflag = 1;
        return;
    }
    else if (adv < NONE_PTC)
    {
        ptcnt = 0;
        ptc_none_beep();
    }
    else
    {
        ptcnt = 1;
    }

    PORTB &= 0xFE; //PTC off
    //FND &= 0x7F;
//PORTD = FND;
    delay(50);

    detectenable = 0;
}

void power_off(void)
{
    FND_RAM[0] = FND_RAM[1] = FND_RAM[2] = FND_RAM[3] = 0xFF;

    powerflag = 0;
}

```

```

        stop();
    }

void stop(void)
{
    PORTB &= 0xFE; //PTC off
    FND &= 0x7F;
    PORTD = FND;

    optime = 0;
    blank_cnt = 0;
    detectenable = 1;

    offtime = 30;
    //stop_bcep();
}

void main(void)
{
    initialize();
    initialize_bcep();

    //power_off();
    FND_RAM[0] = FND_RAM[1] = FND_RAM[2] = FND_RAM[3] = 0xFF;
    powerflag = 0;
    PORTB &= 0xFE; //PTC off
    FND &= 0x7F;
    PORTD = FND;
    optime = 0;
    blank_cnt = 0;
    detectenable = 1;

    enable_interrupt();
}

```

```

while(1)
{
    _WDR();

    if (powerflag -- 1)
    {
        if (stopflag)
        {
            end();
            stopflag = 0;
        }

        if (errorflag)
        {
            error();
            errorflag = 0;
        }

        if (poweroffflag)
        {
            poweroffflag = 0;
            power_off();
        }

        if (levelkey)
        {
            level_up_beep();

            if (++oplevel >= 7)
                oplevel = 1;

            if (!oplevel)
            {
                PORTB &= 0xFE; //PTC off
            }
        }
    }
}

```

```

        FND &- 0x7F;
        PORTD = FND;
    }

    level_display(oplevel);
    ptchugetime = HUGE_TABLE[oplevel];
    delay(10);
}

if (timekey)
{
    if (!optime) //start
    {
        optime = 10;
        start_beep();
        start();
    }
    else if (optime < 31)
    {
        optime += 10;
        time_up_beep();
    }
    else if (optime == 40) //stop
    {
        optime = 0;
        stop();
        stop_beep();
    }
    else
    {
        optime = 40;
        time_up_beep();
    }
}

```

```

        time_display(optime);
        delay(10);
    }

    if (powerkey)
    {
        power_off_beep();
        power_off();
        delay(10);
    }
}
else if (powerflag == 0)
{
    if (powerkey)
    {
        power_on();
        delay(10);
    }
}
}

}

void end(void)
{
    char cnt = 3;

    stop();

    while(1)
    {
        _WDR();

        if (powerkey || timekey || levelkey)
            break;
    }
}

```

```

        if (endbeep)
        {
            end_beep();
            endbeep = 0;
            cnt--; //beep 3회 제한
            if (!cnt)
                break;
        }
    }
}

```

```

void error(void)
{
    long td = 0;

    stop();

    while(1)
    {
        _WDR();

        if (powerkey)
        {
            power_off_beep();
            power_off();
            break;
        }

        if (timekey)
        {
            while(timekey)
            {
                _WDR();
            }
        }
    }
}

```

```

        if (++td > 3000000)
        {
            td = 0;
            time_display(errorflag);
            errordisp = 1;
        }
    }
else td = 0;

if (errorbeep)
{
    error_beep();
    errorbeep = 0;
}
}
}

```