

# ATmega128를 이용한 온·습도 자동제어 모형 제작

김민구\* · 구태희\* · 김영민\* · 심새로얼\* · 소대화\*\*

## Design Clean Room of Controlled Temperature and Humidity using ATmega128

Kim, Min-Gu, Kuh, Tae-Hoi, Kim, Young-Min, Shim, Saero-Eol, Soh, Dea-Wha

**Abstract** - Nowadays, our life is closely related to temperature and the humidity. To control the temperature and the humidity have pursued the comfort environment. We develop that formware is controled temperature and humidity that used the ATmega128. We embody home-automation system to use the ATmega128, temperature sensor, LCD, relay device and peltier device.

**Key Words** : LCD, Atmega128, SENSOR - MCU

### 1. 서론

현재 우리가 살아가고 있는 생활에서 온·습도는 상당히 밀접한 관계를 맺고 있으며, 온·습도 조절을 통하여 쾌적한 공간을 추구하려고 하고 있을 뿐만 아니라 세밀한 공정을 요하는 작업장에서 온·습도는 제품에 큰 영향을 미치고 있다. 그래서 우리는 이런 사회 환경의 흐름을 바탕으로 센서를 통한 현재의 온·습도의 데이터의 값과 기준 값과 비교 및 제어 할 수 있는 C언어 프로그램 개발을 하여 ATmega128를 통한 기본적인 시스템을 만들었다. 센서를 통해 입력된 데이터 값과 기준 값을 LED상으로 표현하였으며 현재의 온·습도장치 동작 상태도 추가적으로 보여주므로 사용자의 인터페이스를 높였다. 온·습도장치의 작동은 데이터 값과 기준 값의 차이가 발생하였을 때 동작한다. 즉 일정온도 이상으로 올라가면 팬을 가동하여 차가운 바람을 유입하

고 습도가 내려가면 가습기를 가동하여 쾌적한 환경을 유지할 수 있도록 하는 것이다. 특히 온도장치는 펠티어 소자를 이용하여 온·냉풍의 방향을 제어할 수 있도록 제작하였고, 습도장치는 데이터 값의 차이에 의한 동작 유·무를 제작하였다. 추가적인 기능으로 데이터 값의 차이가 임계온도로 넘어가면 응급상황으로 판단하여 관리자의 핸드폰으로 응급 메시지를 전달할 수 있도록 하였다.

### 2. 기본구성

홈오토메이션 시스템을 구현하기 위해 우리는 메인보드에 들어갈 ATmega128, 스위칭 역할을 하는 릴레이 소자, 통신을 할 수 있는 통신 회로, 습도 현재 상태를 나타내주기 위한 LCD를 사용하였다.

\* 명지대학교 전자공학과 4학년

\*\* 명지대학교 전자공학과 교수

## 2.1 소프트웨어

### 2.1.1 compile 및 Tool

#### a. WinAVR

WinAVR은 크로스 컴파일로서 AVR GCC를 기반으로 하고 있다.

무료 컴파일러로써 보편화 되어 쉽게 구할 수 있었으며 사용하기에도 쉬웠다. 우리는 WinAVR-20060421버전을 사용하였다.

#### \* Programmers notepad

각종 프로그램을 작성할 수 있는 툴로써 다루기가 용이하였다. Text로 프로그램을 작성 후 c파일로 저장한 후 MFile로 만들어진 make파일을 hex파일로 컴파일 하였다.

#### \* MFile

컴파일을 위해서 make파일을 만드는 툴로써 작성한 프로그램을 오픈해서 MCU를 설정만 하면 되었다.

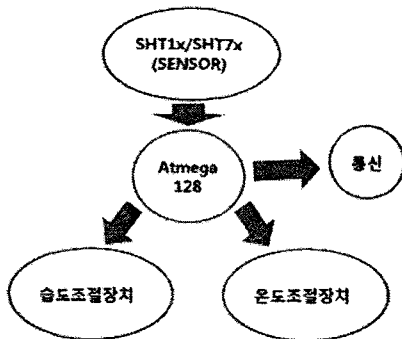


그림1. 전체 시스템 구성도

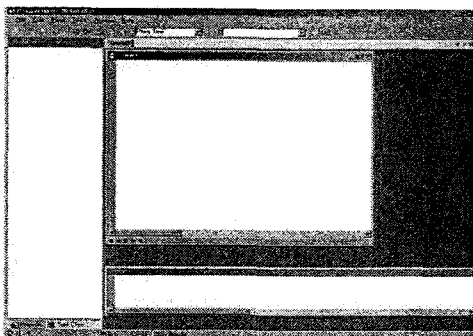


그림2. WinAVR 실행 모습

#### b. PonyProg

PC에서 MCU로 Download를 해주는 툴이다. hex파일이 만들어지면 MCU 선택 후 바로 다운로드를 실행하였다. 처음에 다운로드를 하는데 Device missing or unknown device(-24)라는 경고창이 여러 차례 나타났고, 퓨즈비트를 설정하지 않아서 이러한 에러가 발생하였다.

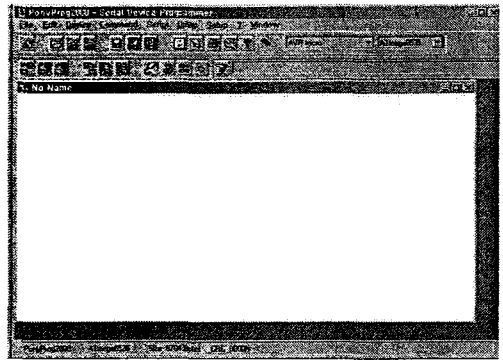


그림3. PonyProg의 실행 모습

## 2.2 하드웨어

홈오토메이션 시스템을 구현하기 위해 우리는 메인보드에 들어갈 ATmega128, 스위칭 역할을 하는 릴레이 소자, 통신을 할 수 있는 통신 회로, 습도 현재 상태를 나타내주기 위한 LCD를 사용하였다.

### 2.2.1 ATmega128

현재 시중에 나오는 많은 AVR중에서 어떤 것을 선택할까 고민 하는 상황에서 ATmega128은 요즘 많이 사용하고 있으며 상대적으로 많은 정보를 얻을 수 있어서 ATmega128을 선택하게 되었다.

ATmega128은 향상된 RISC 구조의 AVR을 기본으로 한 저 전력 CMOS 8비트 마이크로컨트롤러이다. ATmega128은 port A 부터 port G 까지 있으며 2개의 8bit타이머/카운터와 2개의 확장된 16-bit타이머/카운터로 구성되어 있으며

실시간 카운터의 분리된 오실레이터를 가지고 있다.

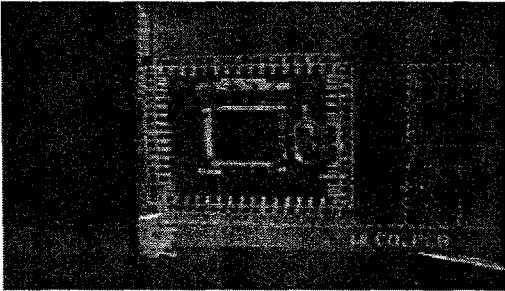


그림4. ATmega 128의 외형

또한 power-on Reset and programmable Brown-out Detection 기능과 RC Oscillator 조정기능을 내장하고 있으며 외부와 내부 인터럽트 소스를 가지고 있는 특징이 있다.

### 2.2.2 LCD 모듈

교육용으로 널리 쓰이는 16\*2 캐릭터 LCD를 사용하였다.

온도·습도상태를 알려주어야 하므로 백라이트가 있는 제품을 사용하였다. 한글은 구현할 수 없지만 우리가 초기에 계획했던 것들을 충분히 디스플레이 할 수 있어서 사용하였다.

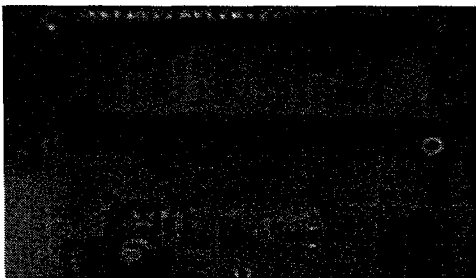


그림5. LCD 모듈

### 2.2.3 팬티어 소자

팬티어 소자는 두개의 서로 다른 금속이 2개의 접점을 가지고 붙어있을 때, 이 두 금속 양단

에 전위차를 걸어주면 전위의 이동에 의해서 열의 이동이 발생하는 현상을 이용한 소자를 말하며 간단한 구조와 환경친화성, 그리고 높은 신뢰성(물리적인 동작 구조를 전혀 가지지 않는 전기회로로만 구성되기 때문에 고장 날 여지가 거의 없다.)을 가지고 있어서 국부 냉각기 등에 널리 사용되고 있는 소자이다.

우리는 이 소자를 온도가 올라가면 팬티어 소자에 전위차를 걸어줘 차가운 바람이 팬을 통해 실내로 들어오게 하고 온도가 일정기준 이하로 떨어지면 따뜻한 바람이 실내로 들어오게 하였다.

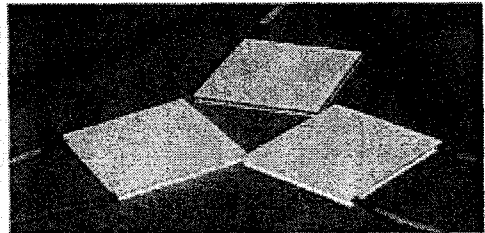


그림6. 팬티어 소자

### 2.2.4 릴레이 소자

MCU에서 나오는 출력전압 5v를 이용해서 12v와 220v의 스위치 역할을 한다.



그림7. 릴레이의 외관

### 2.2.5 통신회로

통신회로 구성을 위하여 CX86500 SCXV.xx

Modem을 이용했다. Cool edit pro를 사용하여 음성정보를 디지털화하고 전화를 이용하여 호출 가능하기 위한 부분이다.



그림8. CX86500 SCXV.xx Modem

### 2.2.6 SENSOR (SHT11)

온도와 습도센서가 내장되어 있는 소자로서 전통적인 습도 센서의 문제점을 해결하기 위해, MEMS (Micro Electro Mechanical System) 기술을 응용한 Finger Electrode System은 정전 용량을 형성하고 그 위에 다양한 보호 층과 폴리머 커버 층을 형성하여 이전까지 구현되지 못했던 방법을 통해 인터페이스로부터 센서를 보호하는 퓨전 센서이다. 이러한 기술들은 기존의 정전 용량 및 저항형 방식의 습도 센서의 단점인 응축 및 침수 시 센서의 수명을 다하는 단점들을 극복함으로써 전통적인 방식의 센서로는 불가능한 완전히 새로운 Application 영역에 대한 가능성을 열어 놓게 되었다.

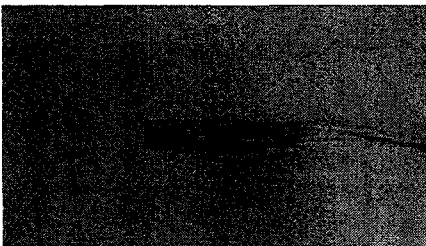


그림9. SHT11 온/습도센서

[커맨드 송신과 데이터 수신]

SHT1x로 클럭을 보냄으로써 데이터를 주고 받는 형식이다.

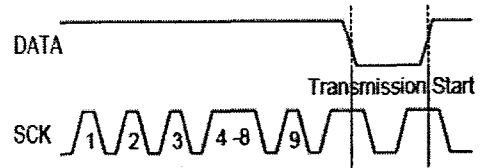
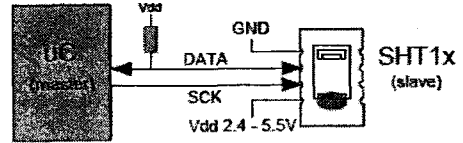


그림10. master/slave간 송수신 및 클럭 파형

데이터 신호가 falling edge와 rising edge에서 클럭이 high일 경우 데이터의 전송이 시작된다. 데이터가 SCK의 1~3 bit는 address bit, SCK의 4~8 bit는 command bit, SCK의 9 bit는 ACK bit이다.

※ soft reset : IC를 리셋하여 스테이터스 레지스터를 기정값으로 초기화하고 다음 커맨드까지 11ms 대기한다. SHTx는 최소 11ms의 딜레이를 요구한다.

1. 전송개시
2. mcu에서 command를 SHT로 송신
3. SHT에서 command 수신 후 ACK 송신
4. SHT에서 온도, 습도의 측정
5. 측정 완료 후 data 라인 low로 만들
6. SHT11이 data 송신 - 8bit 길이로 1 byte 씩 2회로 나누어 송신
7. mcu에서 최초 1byte 수신했을 때 data를 low로 송신
8. 2번째 byte도 반복, CRC 체크 섬이 필요 없을 시 mcu에서 H 송신

※ CRC-8 체크섬 data의 송수신

mcu가 data를 받고 ACK 신호를 보낼 때 low로 했을 경우에는 계속해서 SHT11이 CRC-8의 체크섬 데이터를 송신. 이 data는 LSB(bit 0)에서 MSB(bit 7) 순서로 송신되므로 주의를 요한다.

<표 1> 데이터 값

command	code
reserved	0000x
measure temperature	00011
measure humidity	00101
read status register	00111
write status register	00110
reserved	0101x~1110x
soft reset	11110

온도 데이터를 온도 값(°C)으로 변환

SHT11에서 수신한 온도 데이터 DT의 온도 T°C로의 변환은 다음과 같은 식에 의한다.

$$T = d_1 + d_2 DT$$

d1 및 d2는 표 2, 3과 같이 전원전압과 분해능 bit 수에 따라 선택

표 2 온도변환계수d1

VDD	d1(°C)
5V	-40.00
4V	-39.75
3.5V	-39.66
3V	-39.60
2.5V	-39.23

습도 데이터를 상대습도(%)로 변환

습도 data DH의 상대 습도 H%로의 변환은 다음 식에 의한다.

$$H = c1 + c2DH + c3DH^2$$

c1 ~ c3는 표 4와 같이 분해능 bit 수에 따라 선택

<표 3> 온도변환계수d2

분해능	d2(°C)
14 bit	0.01
12 bit	0.04

<표 4> 상대습도변환계수

분해능	c1	c2	c3
12 bit	-4	0.0405	-2.8×10-6
8 bit	-4	0.648	-7.2×10-4

2.2.7 PAN

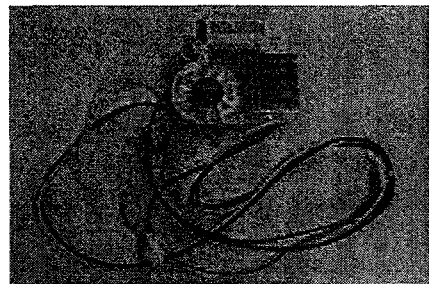


그림11. 팬티어 소자의 방열판

팬티어의 양면에 팬의 방열판을 접착시켜 HITTING과 COOLING을 함으로써 공기를 순환시키게 해주는 장치이다. 우리가 필요한 온도 조절 장치로는 팬티어의 한쪽 면과 하나의 팬이 사용되지만 팬티어의 반대쪽과 나머지 팬은 팬티어의 손상을 막는 HITTING과 COOLING역

할을 한다.

### 3. 제작과정

#### 3.1 기본구성

① 기본 외형은 아크릴로 제작, 온도구동장치는 펠티어 소자를 이용한 팬으로 제작, 습도구동장치는 가습기로 대체

② 높이(cm)X가로(cm)X세로(cm)의 내부에 작업 모형공간과 습도 구동장치로 구성, 외부에 온도 구동장치 연결

③ 작업 공간 내에 온·습도센서설치 및 메인보드에 연결

④ 작업 공간 외부에 LCD, 제어버튼, 통신단자 BOX 제작 후 설치, 메인보드에 연결

⑤ WinAVR 사용하여 프로그램을

#### 3.2 기본외형제작

① 전체외형

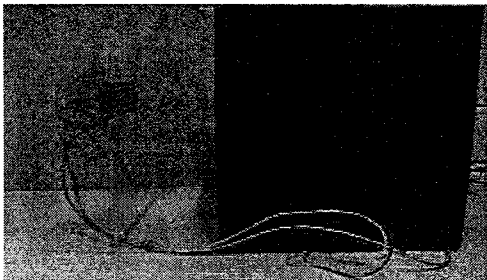


그림12. 온/습도 조절 공간의 외형

가상의 작업공간에 외부의 공기를 유입하여 온도를 조절할 수 있는 온도조절장치를 설치, 내부에는 작업공간의 습도를 조절할 수 있는 습도 조절장치를 설치한다.

② LCD & 컨트롤 BOX

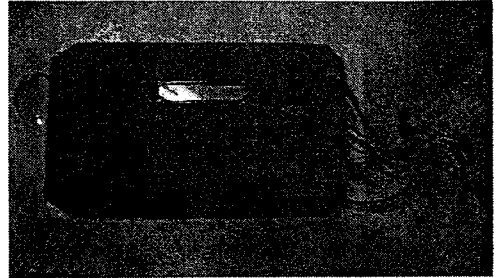


그림13. LCD 표시장치와 컨트롤 BOX

LCD를 통하여 현재의 온·습도를 알 수 있으며 현재 작동중인 온·습도장치의 동작 상태를 파악할 수 있다. 제어 컨트롤은 5개의 버튼으로 구성되어 있다.

(1) RESET : 동작 상태를 초기상태로 되돌린다.

(2) SET : 기준 값의 온·습도를 설정하기 위한 상태로 간다.

(3) UP : 온도와 습도의 수치 값을 올린다.

(4) DOWN : 온도와 습도 수치 값을 내린다.

③ 메인보드

Atmega128을 중심으로 SHT1x/SHT7x의 센서로부터 들어온 데이터 값을 기준 값과 비교하여 릴레이소자 5개를 제어한다.

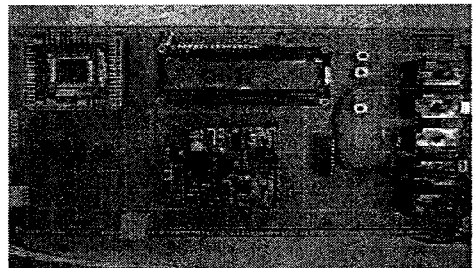


그림14. 메인보드

1번째 릴레이 소자는 펠티어 소자에 전원을 공급한다. 2,3번째 릴레이 소자는 펠티어의 극성을 서로 바꾸는데 사용한다. 4번째 릴레이 소자

는 온도조절장치의 팬을 구동시키기 위한 전원을 공급한다. 5번째 릴레이 소자는 습도조절장치의 220v전원의 공급, 차단을 한다.

회로의 중앙에 놓인 통신회로는 Atmega128에 저장되어 있는 기준 값과 센서를 통해 들어온 데이터 값이 임계수치를 넘었을 경우 통신 단자를 통해 저장되어 있는 전화번호로 위험을 알려준다.

## 4. 동작과정

① LCD & 컨트롤 BOX에서 SET버튼을 누른 후 기준 값 설정상태 화면으로 이동 후 UP, DOWN버튼을 이용하여 기준 값을 설정해준다.

② 설정상태 화면에서 아무런 입력이 없고 일정시간이 지나면 현재 상태화면으로 이동한다.

③ 센서들 통해 들어온 데이터 값이 기준 값에 비해 온도가 낮고 습도가 낮다면 온도조절장치가 작동하여 뜨거운 바람이 유입되고 습도조절장치가 작동하여 습기가 작업공간으로 유입한다.

④ 센서를 통해 들어온 데이터 값이 기준 값에 비해 온도가 높고 습도가 높다면 온도조절장치의 팬티어 소자의 극성이 바뀌고 차가운 바람이 유입되며 습도조절장치는 릴레이 소자에 의하여 전원공급이 차단되어 작동을 멈춘다.

⑤ 센서를 통해 들어온 데이터 값과 기준 값이 차이가 없다면 온·습도 조절장치는 작동을 멈추고 계속적으로 들어온 데이터 값과 비교하면서 대기한다.

⑥ 작업공간의 온도가 임계온도를 넘어갈 경우 연결된 전화선을 통하여 Atmega128에 저장되어 있는 전화번호로 긴급메시지를 전달한다.

## 5. 결론

온·습도 제어 시스템의 원리와 구조를 이해함과 동시에 실생활에서 사용하고 있는 시스템과 근접한 작품제작을 목표로 길지 않은 시간동안 최선의 노력을 하였다. 계획한 목표 달성을 위해 많은 어려움이 있었지만, 우리는 ATmega128에서 지원하는 기능을 통해 온·습도 제어시스템과 접목시켜 구현하였다. 본 프로젝트 수행을 통하여 팬티어 소자의 구체적 내용을 이해할 수 있었으며, 릴레이 소자의 on/off 스위치 역할과 기능으로부터 온·습도제어공간의 시스템을 구현하여 실험적 수행 결과를 확인하였다.

하지만 팬티어 소자의 열이 팬에 전도되는 과정에서 열 손실과 관련된 문제점과 모형 작업장의 구조적인 문제로 인해서 기대했던 만큼의 만족스러운 결과를 나타내지는 못하여 아쉬움이 남는다.

현재의 온·습도 제어 시스템은 여러 분야에서 사용되고 있으므로 본 프로젝트의 경험은 향후 활동에 많은 도움이 될 수 있을 것이다.

### 참고문헌

- [1] 이상철, "마이크로프로세서 기초 실습", 한울 출판사, 2006.
- [2] 이재창, "로봇스터디 ATmega128", 동일출판사, 2006.
- [3] 김용석, "AVR설계 및 프로그램", 동일출판사, 2004.
- [4] 최선영, "C언어 프로그래밍", 과학기술, 2002.
- [5] <http://blog.naver.com/ds5pnz> AVR Essay
- [6] <http://cafe.naver.com/carrotty>