****

|  |  |
| --- | --- |
| 최종 작성일 | 2018.12.11 |
| 작성자 | 공찬형 |
| 학번 | 20133950 |

**Predicting Time Required to Wake Up Machine Using Temperature, Altitude, Light and Pressure Data**

Linux Term Project Report

**목차**

[1. Abstract 2](#_Toc532348289)

[2. Introduction 3](#_Toc532348290)

[2.1 연구 배경 및 동기 3](#_Toc532348291)

[2.2 설계 목표 3](#_Toc532348292)

[3. Development 3](#_Toc532348293)

[3.1 설계 상세 3](#_Toc532348294)

[3.2 구현 6](#_Toc532348295)

[4. Result 10](#_Toc532348296)

[4.1 구현 결과 10](#_Toc532348297)

[5. Conclusions 11](#_Toc532348298)

[5.1 결론 11](#_Toc532348299)

[5.2 한계점 11](#_Toc532348300)

[6. References 12](#_Toc532348301)

[7. Appendix 13](#_Toc532348302)

# Abstract

일반적으로 인간은 하루 일과의 30~40% 이상을 수면하므로, 수면 환경 조건은 사람의 건강, 쾌적 및 하루의 작업 능률에 많은 영향을 미치게 된다. 따라서 인간이 쾌적한 수면을 취했다고 느낄 수 있는 환경을 조성하는 것은 중요하다. 인간의 주위 환경으로부터 쾌적한 상태를 느끼는 연구는 상당히 진행되어 왔으며, 일반적으로 온도와 습도에 가장 크게 영향을 받는 것으로 결론지어진다. 또한 최근 사물인터넷 기술의 발전과 함께 인간의 주위 사물들에 다양한 센서들을 장착한 초소형 컴퓨터가 탑재되어 있으며 수집되는 정보를 바탕으로 인간의 거주 환경을 최적의 상태로 유지하려는 시도가 있어 왔다. 따라서 초소형 컴퓨터를 이용하여 주위 환경이 인간의 수면의 질에 미치는 영향을 분석한다면, 가장 높은 질의 수면을 취할 수 있는 수면 환경을 파악하고 그 환경을 조성할 수 있을 것이다. 본 연구 및 개발은 인간의 수면의 질을 보편적으로 알 수 있는 척도인 기상 소요 시간에 수면 장소의 온도, 기압, 광량 및 고도가 미치는 영향을 분석하여 인간의 수면의 질을 높이고, 특정 광량 상태에 기상 시간을 제어하는 기상 환경 조성 사물 인터넷 기기를 제시한다.

# Introduction

## 연구 배경 및 동기

라즈베리파이 및 아두이노로 대표되는 초소형 컴퓨터가 값싼 가격에 보급됨에 따라, 일반적인 가전 제품이나 수동 제어 장치를 인터넷에 연결하여 제어할 수 있게 되었다. 특히 초소형 컴퓨터는 에어컨, 난방기 혹은 제습기 등에 설치되어 상시 같은 환경을 조성하도록 스스로를 제어한다. 하지만 현재 시판되는 사물인터넷 환경 조성 가전제품에는 광량을 측정하여 수면 환경을 제어하지 않기 때문에, 수면 상태로 들어간 인간의 체온 혹은 습도에 대한 민감성 변화에 대응할 수 없다. 이 민감성으로 대변되는 쾌적함은 광량에 밀접할 수 밖에 없는데, 이는 인간의 수면 혹은 활동 상태에 광량이 상당한 영향을 미치기 때문이다. 이에 본 연구는 광량을 기반으로 사용자의 수면 활동을 제어하며 기상 지연 시간을 바탕으로 주위 환경을 분석하여 수면의 질을 높이는 방법을 제시한다.

## 설계 목표

본 연구 및 개발은 주위 환경 정보를 수집하고 수집된 광량 정보를 설정된 기상 광량에 비교하여 사용자의 기상 시간을 제어하고, 정보를 모니터링 및 분석할 수 있는 기기를 개발하는 것을 목표로 한다. 기기는 High Precision AD/DA 보드에 부착된 가변저항센서 및 조도 센서와 추가적으로 설치된 온도 센서를 기반으로 정보를 수집하며 수집된 정보를 영구적으로 저장하며, 기간에 따라 조회할 수 있어야 한다. 또한 수집된 정보를 바탕으로 GPIO 장치를 제어하여 부저를 울려 사용자를 기상시키고, 기상에 걸리는 시간을 수집할 수 있어야 한다. 최종적으로 수집된 정보를 데이터베이스에 영구적으로 저장하고, 웹 기반 UI를 통하여 조회 및 분석할 수 있어야 한다.

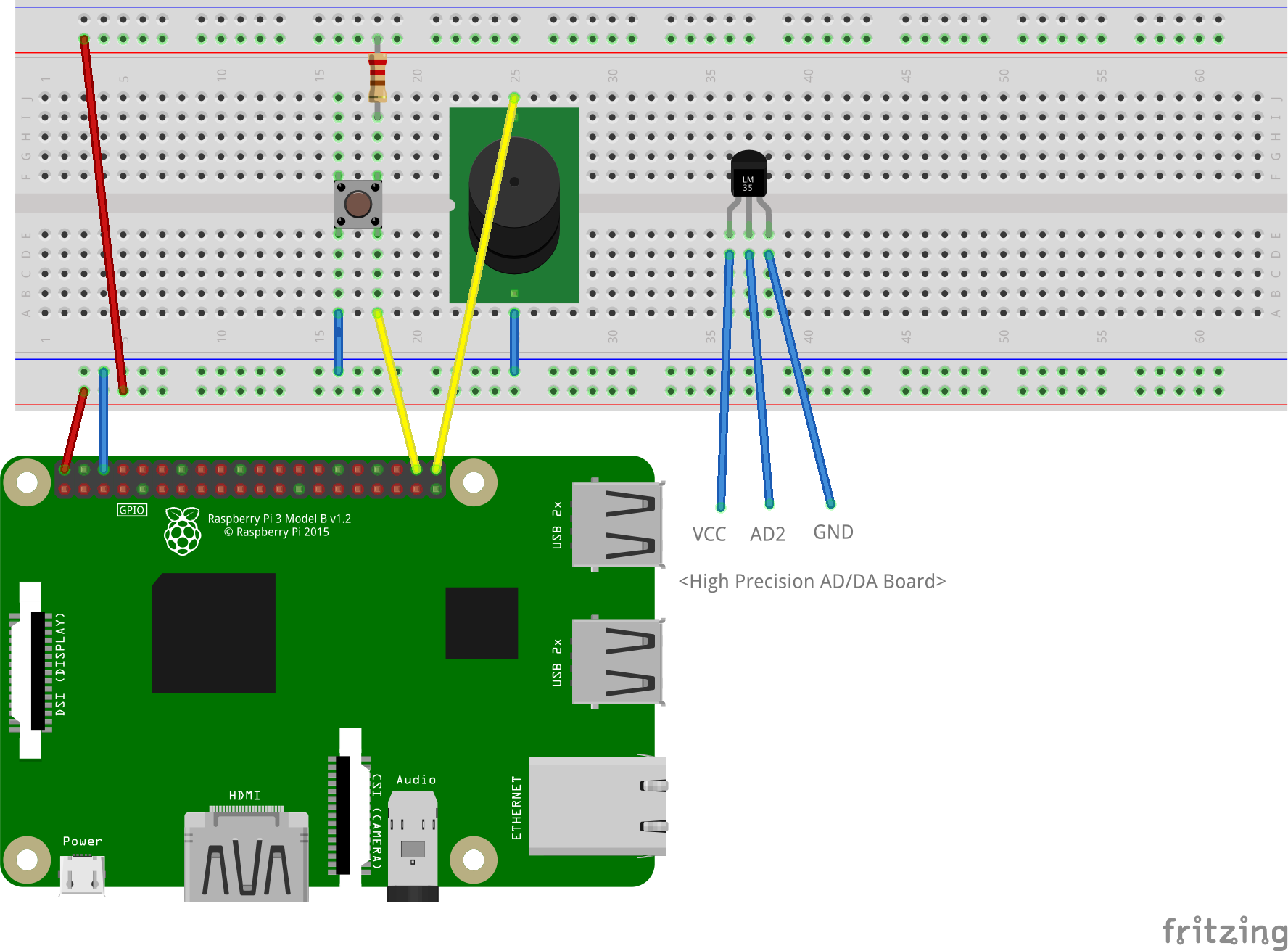
# Development

## 설계 상세

1. 하드웨어 설계

수면 환경의 정보를 수집하기 위하여 High Precision AD/DA 보드를 사용한다. 추가적으로 온도 정보를 측정하기 위하여 라즈베리파이 브레드보드에 온도 센서 (LM35DZ) 를 부착한다. 부착된 센서는 디지털 센서이기 때문에, 전송되는 전압 정보를 아날로그 정보로 변경하기 위하여 High Precision AD/DA 보드의 아날로그 단자에 연결한다. 본 연구에서는 변환 단자 2번을 사용하여 정보를 수집하였다. 또한 기상 알람을 시작할 광량을 제어할 수 있도록 하기 위하여 High Presion AD/DA 보드에 설치된 가변 저항 센서를 이용한다.

기상을 제어하는 부저와 부저를 제어하고 기상 시간을 기록할 장치를 위하여 라즈베리파이와 GPIO 단자를 사용한다. GPIO 단자는 추가된 High Precision Board의 단자를 사용한다. 기상 알람을 위한 부저를 설치하고 GPIO 29번 단자를 이용하여 제어할 수 있도록 회로를 구성한다. 부저를 제어하고 기상 시간을 기록할 수 있도록 GPIO 28번 단자를 이용한다. 버튼의 + 회로에 1000 옴 저항을 설치하고, 버튼이 눌렸을 때에 접지를 위하여 접지 단자를 연결한다. 최종적으로 구성된 회로의 모습은 <그림 3-1>의 모습과 같다.



<그림 3-1>

1. 소프트웨어 설계

소프트웨어는 하드웨어를 제어하는 모듈, 주위 환경 정보를 수집하는 모듈, 실시간 정보를 표시할 수 있는 모듈과 데이터를 데이터베이스로 전송하는 모듈로 구성된다. 추가적으로 데이터베이스를 읽어 웹 UI로 제공할 수 있는 프로그램이 추가적으로 동작한다.

하드웨어를 제어하는 모듈은 2가지의 역할을 담당한다. 클라이언트의 요청으로부터 GPIO에 데이터를 기록하여 부저를 울리고, 지속적으로 버튼의 데이터를 읽어 버튼의 눌림에 반응하여 부저를 끌 수 있도록 해야 한다.

주위 환경 정보를 수집하는 모듈은 High Precision AD/DA 보드의 I2C 통신으로부터 전송된 조도 및 가변저항 센서, 온도 센서로부터 수집된 전압 센서를 읽는 역할을 담당한다. 해당 센서정보 수집 모듈은High Precision AD/DA 보드와 함께 포함된 소스 코드를 정제하여 사용한다.

실시간 정보를 표시하는 모듈은 커널에서 사용하는 소프트웨어 타이머를 이용한다. 특정 주기에 맞추어 설정된 함수를 실행시킬 수 있도록 한다. 또한 데이터의 날짜별 분류와 측정 시간을 위하여 시스템 시간을 불러오는 역할을 한다.

데이터베이스 모듈은 수집된 데이터를 데이터베이스로 전송하는 역할을 한다.

웹 모듈은 사용자의 특정 검색 범위 내의 로그를 검색하고 기상 시간에 대한 분석 정보를 보여주는 기능을 제공한다. 구현된 소프트웨어와 별개로 수집된 데이터베이스와 통신하며 수집 기간을 바탕으로 검색하고, 특정 수면 환경에서의 검색을 통하여 평균 기상 소요 시간을 분석한다.

전체적인 모듈 구성도와 역할 분배 개요는 <그림3-2>와 같다. 소프트웨어의 기본 구조는 제공된 프로젝트 기본 구성을 따르되, 추가적으로 시간에 관련된 모듈을 사용하기 위하여 Timer 모듈을 추가하였다. 데이터베이스 제어 모듈은 Util 디렉토리에 추가하여 설계하였다.



<그림 3-2>

## 구현

센서 모듈에서 주위 환경에 대한 정보를 수집하기 위하여 High Precision AD/DA 보드에서 제공하는 AD 컨버터를 제어하기 위한 ADS1256 소스 코드를 라이브러리로 작성하였다. 소스 코드를 바탕으로 센서를 테스트해 보았을 때에, 일반적인 사무실 환경의 조도는 200~300 mV로 측정되었으며, 소등 상태는 2300~2700mV로 측정되었다. 가변 저항의 측정 범위는 10~3300mV로 측정되었다. 이 측정량을 통하여 다양한 사용자의 수면 환경, 램프를 켜두는 상황이나 조광량을 고려하여 가변 저항 센서를 통하여 기상 환경의 조광 임계치[[1]](#footnote-1) 를 설정하여 기상 환경 조광량을 조정할 수 있도록 하였다. 추가적으로 설치한 온도 센서의 온도와 저항간 변환 공식을 도출하기 위하여 부분 실험을 실시하였고, 결과는 다음 <표3-1> 과 같다.

|  |  |  |
| --- | --- | --- |
|  | 측정 데이터 | 실내 온도 |
| #1 | 950728 | 26.2’C |
| #2 | 1003753 | 7.7’C |
| #3 | 982456 | 13.2’C |

<표 3-1>

데이터 변환을 위하여 측정 데이터와 실내 온도는 선형 일차방정식을 따른다고 가정하였다. 측정된 데이터로부터 얻어진 수식은 <수식 3-1> 과 같다.

*온도값 = 측정 데이터 \* -3.488 \* 10-4 + 357.9*

<수식 3-1>

비교군 실험을 실시하여 오차 범위를 계산하였을때, 비교 대상인 Pioneer 600 보드와 약 3.6% 오차율을 보였다. 추가적으로 프로젝트의 요구사항인 고도 및 기압 정보는 사용하는 High Precision AD/DA 보드 내에 설치되어 있지 않으며 보급되지 않았기 때문에 더미 데이터를 사용하여 데이터베이스 및 파일에 기록하였다.[[2]](#footnote-2)

모니터 모듈은 라즈베리파이 터미널 환경에서 간단한 텍스트 출력을 지원하도록 구현하였다. 구현된 모니터 모듈은 수집 센서 혹은 수집 단위에 종속되지 않게 하기 위하여 센서의 데이터를 표현하는 구조체를 통하여 출력된다. 구조화된 센서 데이터는 데이터의 이름, 데이터의 값, 데이터의 단위로 구성되며 구조체 생성시에 동적으로 메모리를 할당하도록 하였다. 모니터 모듈은 매 센서를 통한 측정시마다 수집되어 구조화된 데이터를 기반으로 가변 저항 센서, 광량, 온도, 고도 및 기압을 출력한다.

로거 모듈은 수집된 데이터를 csv형식[[3]](#footnote-3)의 파일로 출력할 수 있도록 구현하였다. 또한 각 파일의 이름을 날짜 형식으로 저장할 수 있도록 하였다. 저장 형식은 2018\_12\_10\_log.dat 와 같이 연\_월\_일로 구분되어진다. 타이머 모듈로부터 습득하는 날짜 정보다 리눅스 타임에 기반하고, 와이파이 환경에서의 라즈베리파이 안정성을 위하여 시간 측정이 영국을 기준으로 하기 때문에 실제 시간과 날짜는 측정 장소와 상이할 수 있다. 라즈베리파이 환경설정을 통하여 시간 측정 지역을 한국으로 변경하면 실제 측정 장소와 시간을 동기화 할 수 있다.

타이머 모듈은 리눅스에서 제공하는 Realtime 라이브러리를 이용하여 소프트웨어 타이머를 이용하여 구현하였다. 라이브러리 형식으로 작성하여 모듈 사용시 timer\_init을 통하여 받아온 타이머 ID에 함수와 시간을 기입하여 동작을 시작할 수 있도록 작성하였다. 클라이언트 프로그램에서 선언된 함수와 시작 시간, 실행 간격을 설정하면 선언된 함수를 매 실행 간격마다 실행시키도록 할 수 있다. 본 연구에서 사용되는 클라이언트 프로그램에서는 타이머 모듈을 통하여 매 초마다 측정된 광량을 바탕으로 현재 시간대를 파악하고, 광량에 변화에 어떻게 반응할 지를 판단한다. 또한 측정된 센서 정보를 데이터베이스에 쿼리로 전송하도록 한다.

데이터베이스 모듈은 라즈베리파이에 설치된 데이터베이스에 접근하여 쿼리를 전송하여 데이터를 기록하는 역할을 한다. 타이머 모듈과 마찬가지로 라이브러리 형식으로 작성되어 있다. 프로그램의 특성 상 한번 연결된 데이터베이스에 지속적으로 연결관계를 맺고, 더불어 같은 데이터베이스에 중복되는 연결 시도의 지연시간이 시스템에 전반적으로 영향을 미칠 수 있기 때문에 하나의 데이터베이스에만 연결될 수 있도록 작성하였다. 해당 모듈을 사용하기 위해서는 데이터베이스의 서버, 유저, 비밀번호와 데이터베이스 이름을 통하여 연결을 받고, 테이블 명을 통하여 쿼리를 전송할 수 있도록 하였다. 쿼리의 전송은 센서 데이터를 전송하는 역할을 하기 위하여 SQL Insert 구문을 활용하여 구현하였다.

구현된 모듈을 토대로 클라이언트 프로그램은 다음과 같이 동작한다. 프로그램 진입 시 데이터 구조체와 ADS1256 신호 변환기, GPIO 핀 번호와 데이터베이스를 초기화한다. 이후 매 초마다 반복될 함수를 지정하고 데이터 수집을 시작한다. 초기화 이후 데이터 수집은 매 초마다 이루어지며 데이터를 수집하여 구조화하고, 터미널 상에 출력한 뒤 파일에 해당 내용을 기록하며 수집 데이터값을 클라이언트 프로그램의 전역 변수에 할당하는 과정을 반복한다. 매 초마다 반복하도록 설정된 함수는 전역변수로 할당된 시스템 시간을 데이터베이스 DATETIME 형식으로 변환하고, 현재 광량과 설정된 루프 임계치를 바탕으로 현재 시간대를 파악한다. 여기서 루프 임계치란, 매 루프마다 증가하는 루프값을 통해 현재 시간대를 파악하는 기준을 의미한다. 수면 환경에서의 순간 점등이나 플래시에 알람을 울리지 않도록 하기 위하여 설정하였으며, 가변 저항 장치를 통하여 설정된 임계 광량 미만의 값이 지속되어 증가된 루프값이 루프 임계치를 초과하였을 때에 현재 시간대를 아침으로 판단하여 사용자를 기상시키기 위해 부저가 작동한다. 현재 약 10초정도의 임계치를 두어 설정된 시간 이상으로 임계 이상의 광량 측정이 지속될 시에 아침으로 판단할 수 있도록 하였다. 이어서 현재 시간대가 아침으로 판단 될 시에 부저를 울리고, 버튼의 입력 상태를 파악한다. 버튼이 눌리지 않으면 지속적으로 부저를 울리고, 버튼이 눌리면 부저를 종료한다. 부저가 종료되면 현재 버튼이 눌린 시간을 기록하여 알람 시작 시간, 알람 종료 시간, 측정된 온도, 기압, 광량, 고도를 데이터베이스로 전송한다. 이와 별개로 함수 종료 이전에 매 초마다 발생한 센서 정보 또한 데이터베이스에 측정 시간과 함께 전송한다.

데이터베이스의 데이터 전송과 센서 측정을 각기 다른 함수에 작성하였기 때문에, 파일 내에 기록된 데이터의 시간과 데이터베이스에 기록된 시간이 다를 수 있다. 하지만 두 역할을 분리하면 크게 두 가지의 이점을 얻을 수 있다. 첫째, 예기치 못한 데이터베이스 접속 실패가 시스템 전체에 영향을 미치지 않는다. 이는 데이터베이스 모듈에서의 연산 결과가 측정 및 파일로의 기록에 종속되지 않기 때문이다. 둘째, 데이터베이스를 사용하지 않거나, 부저 혹은 버튼이 배제된 측정 장치로써의 재활용이 가능하다. 현재 하드웨어에 부착된 GPIO 장치나, 연동된 데이터베이스를 삭제한다고 하여도 측정 장치로써 정상적으로 작동하도록 구분되어 있기 때문이다.

전체적인 클라이언트 프로그램에서의 작동 순서는 다음 <그림 3-3>[[4]](#footnote-4) 과 같다.

클라이언트 프로그램으로부터 수집된 데이터를 모니터링하기 위하여 추가로 웹 모듈을 구현하였다. NodeJS 기반으로 라즈베리파이에서 웹 서버를 서비스할 수 있도록 하였다. 웹 서버는 NodeJS의 Express 프레임워크를 기반으로 작성하였다. 또한 웹 페이지는 클라이언트에서 지속적으로 기록하는 데이터베이스와 연동하여 POST 방식을 통하여 설정된 기간 내의 데이터를 조회할 수 있도록 구현하였으며, 일반적인 로그를 볼 수 있는 버튼과 기상 알람 시간과 종료 시간을 볼 수 있는 버튼을 구현하여 일반적인 로그를 보거나 기상 기록을 볼 수 있도록 하였다. 구현된 웹 페이지의 모습은 < 그림 3-4>, <그림 3-5>, <그림 3-6>와 같다.

# Result

## 구현 결과

위 구현 방법을 토대로 시스템을 구현한 결과 매 1초 혹은 2초 단위로 센서 정보가 수집되어 데이터베이스에 저장된 후, 웹 페이지를 통하여 모니터링 할 수 있었다. 수집된 센서 정보를 바탕으로 시간 별 실내 환경을 측정할 수 있었다. 특히 약 5일간 측정된 수면 환경 정보로부터 광량과 온도가 인체의 수면 리듬에 미치는 영향을 분석할 수 있었다. 측정된 데이터는 < 그림 3-4~5 > 에 보여진다. 그림에서 볼 수 있듯이 실시간으로 측정되는 데이터를 웹 페이지를 통해 모니터링 할 수 있으며, 기상 시간을 선택하여 볼 수 있다. 또한 특정 수면 환경에서의 평균 기상 지연 시간을 산출하여 기상 환경을 조성할 수 있다. 또한 측정된 데이터는 데이터베이스 뿐만 아니라 디렉토리 내부의 csv 파일 형태로 저장되며, 각 행마다 가변 저항값, 광량, 온도, 기압, 고도 순으로 기록되어 있다. 표시된 모습은 <그림4-1>에서 확인할 수 있다. 터미널 상에 측정된 데이터를 모니터링 하는 모습 또한 <그림4-2>에서 확인할 수 있다.

결과적으로 구현된 기기는 주위 환경의 온도, 광량, 기압, 고도를 측정하여 기록하여 데이터베이스에 저장할 수 있고, 광량을 통하여 기상시간을 추정해 사용자를 알람을 울림으로써 기상시키며 기상하는데 걸리는 소요시간을 저장하고, 기상 소요 시간과 수면 환경의 상관관계를 분석할 수 있는 웹 기반 UI를 제공한다. 완성된 하드웨어의 모습은 문서 하단 Appendix <별첨 1>에서 확인할 수 있다.

완성된 시스템은 하단 링크를 참조하여 다운받을 수 있다.

*https://github.com/Syndra/Linux\_ADDA*

# Conclusions

## 결론

본 연구는 사람의 수면의 질과 수면 환경의 상관관계를 분석하고, 분석된 데이터를 토대로 스스로 수면 환경을 제어할 수 있는 사물인터넷 기기의 기초가 될 수 있는 수집 및 분석 도구를 개발하는 것을 목표로 하였다. 개발 및 구현 결과를 통하여 수집되는 센서 데이터를 통하여 사용자를 기상시키며 버튼을 통해 사용자가 알람을 종료할 때 까지 걸리는 시간을 통해 각종 수면 환경과 수면의 질의 연관관계를 분석할 수 있는 시스템을 완성하였다. 개발된 시스템은 차세대 초소형 디바이스에서 인간의 수면 환경을 동적으로 판단하여 제어할 수 있는 기기의 제반을 마련하여 실용성을 입증하였다는 것에 그 의의가 있다. 이에 추가적으로 주변 에어컨, 가습기 등을 제어하는 모듈을 부착하여 분석된 데이터를 통하여 수면 환경을 동적으로 개선한다면, 진행된 관련 연구 결과와 더불어 사람의 수면의 질을 높여 인간의 삶의 쾌적함에 크게 기여할 수 있을 것으로 보인다.

## 한계점

본 연구는 크게 두가지의 한계점을 지닌다. 첫째, 라즈베리파이 특성상 과도한 트래픽을 소화하지 못하기 때문에, 웹 기반 UI를 다수의 서버에서 접근 시도한다면, 시스템 전체가 불안정해지거나 데이터 수집 시스템의 실행 주기가 엇나갈 수 있다. 이는 라즈베리파이 내부에서 동작하는 데이터베이스를 외부로 추출하고, 웹 호스팅 서버 또한 추출하면 해결할 수 있다. 둘째, 조도 센서를 통하여 수집된 빛 계측량에 따라 사용자를 기상시킬 때, 가변 저항 장치를 이용하여 적절한 광량 임계치를 설정하는 과정이 많은 시간이 소요된다. 또한 계절에 따라 아침에 기록되는 광량이 달라지기 때문에, 사용자는 반드시 이를 인지하고 광량 임계치를 설정하여야 한다.

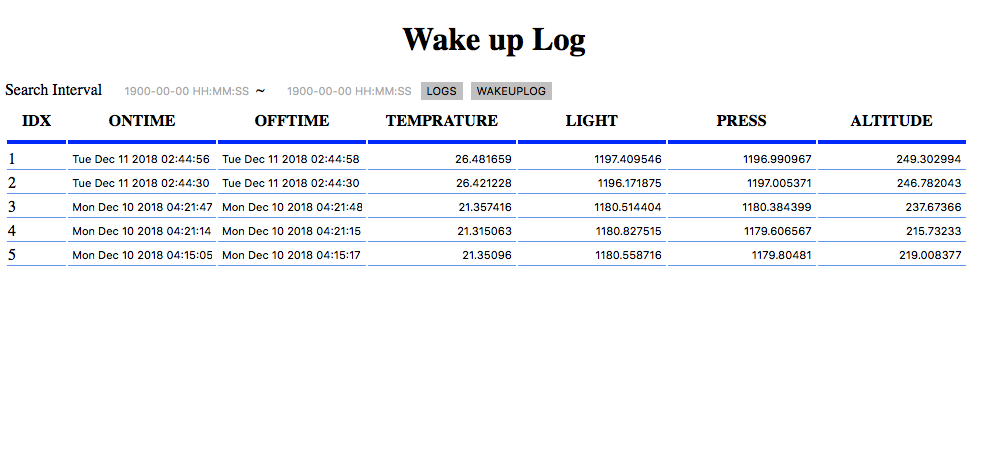
# References

1. 김세환, 이성, 김동규 (2007). 겨울철 온도 및 습도변화에 따른 온열쾌적감에 관한 연구, 설비공학논문집, 19(11), 803-809.
2. 신성윤, 신광성, 이양원. (2010). 센서를 이용한 수면 환경 개선. 한국정보통신학회논문지, 14(11), 2485-2490.

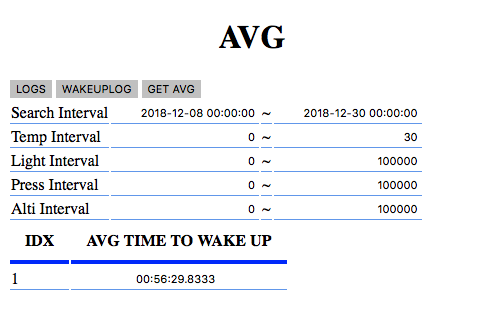
# Appendix



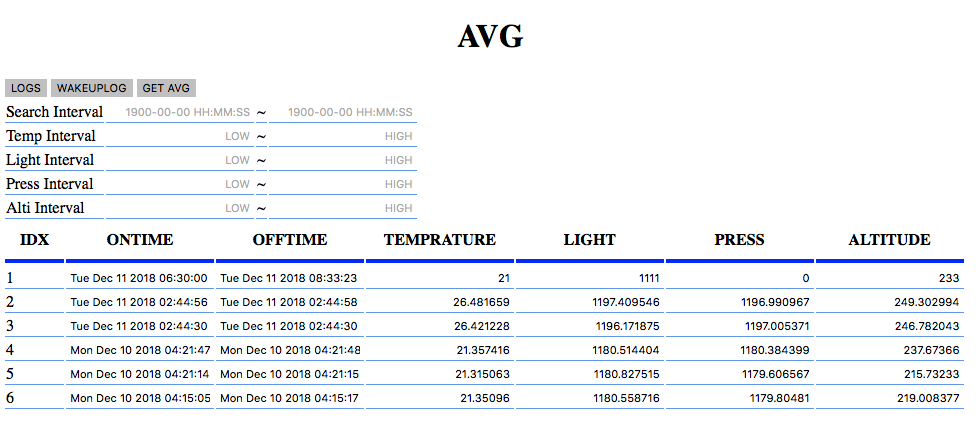
<그림 3-3>



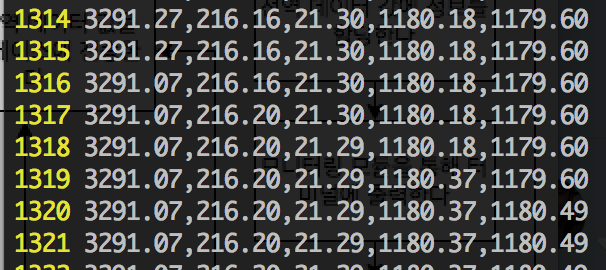
<그림 3-4>



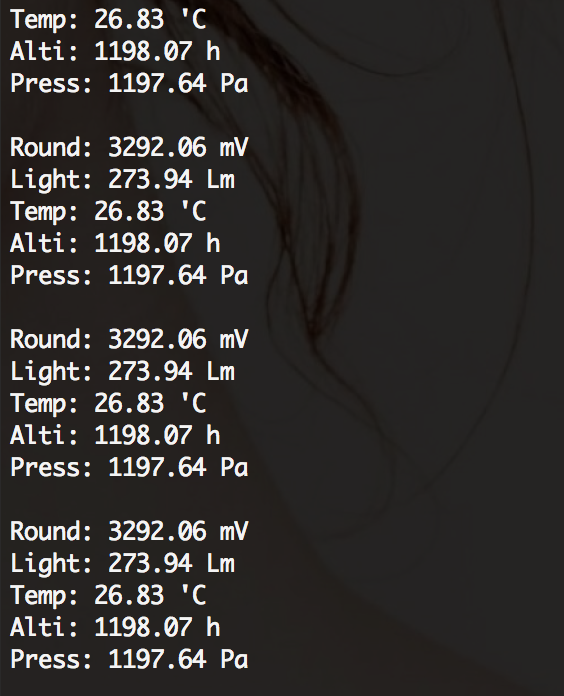
<그림 3-5>



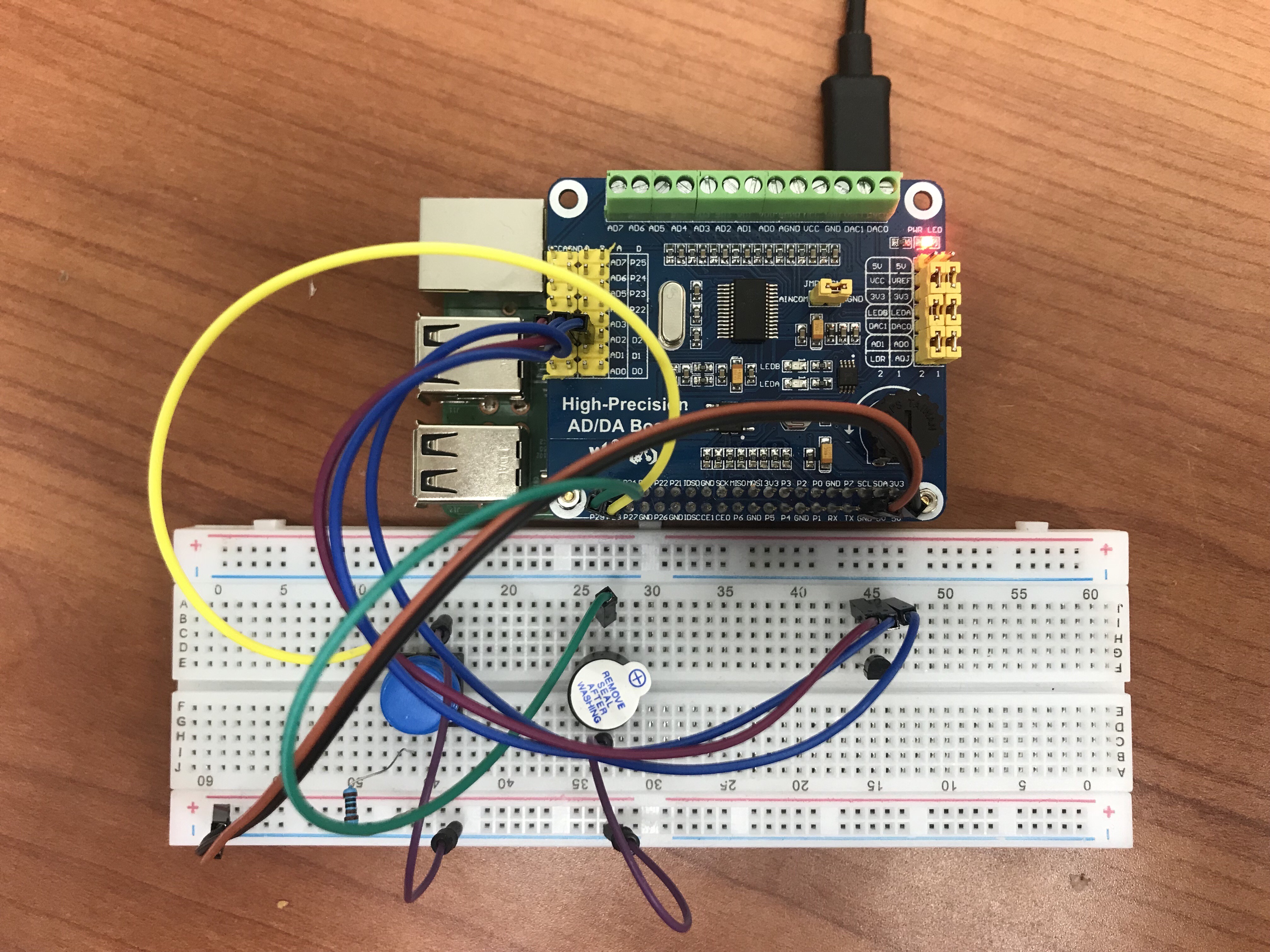
<그림 3-6>



<그림 4-1>



<그림 4-2>



<별첨 1>

1. 아침의 조광량을 판단하는 기준. 예를 들어, 가변저항이 500mv 로 설정되어 있을때, 500mv이하의 센서 데이터 수집시 낮으로 판단하고 부저를 울린다. [↑](#footnote-ref-1)
2. AD 변환기로부터 전송되는 기본 값인 1190~1200mV 를 각 단위에 맞춘 h, Pa 에 맞추어 출력하였다. 따라서 실제 환경의 고도 및 기압과 크게 다를 수 있다. [↑](#footnote-ref-2)
3. 쉼표와 줄바꿈으로 구분되는 형식 [↑](#footnote-ref-3)
4. 7.Appendix 참조 [↑](#footnote-ref-4)