**Project** ‘Green Farm’



**팀 이름: Makers**

**오세형**

**E-MAIL :** [**ohhsehyung2@naver.com**](mailto:ohhsehyung2@naver.com)

**☏ : 010-4795-7694**

**김태희**

**E-MAIL :** [**ktkzzabg@naver.com**](mailto:ktkzzabg@naver.com)

**☏ : 010-6700-4810**

**이정우**

**E-MAIL :** [**woo601@gmail.com**](mailto:woo601@gmail.com)

**☏: 010-4757-9297**

**한국직업능력교육원 안산**

Contents

[**1.** **개요** 1](#_Toc80091931)

[**1.1 프로젝트 명** 1](#_Toc80091932)

[**1.2 프로젝트 기간** 1](#_Toc80091933)

[**1.3 프로젝트 목적** 1](#_Toc80091934)

[**1.4 프로젝트 관련 국내외 동향** 1](#_Toc80091935)

[**2. 프로젝트 범위** 2](#_Toc80091936)

[**2.1 작업명세서** 2](#_Toc80091937)

[**3. 시스템 구축 환경** 2](#_Toc80091938)

[**3.1 하드웨어 구축** 2](#_Toc80091939)

[**3.2 소프트웨어 구축** 2](#_Toc80091940)

[**4. 프로젝트 추진체계** 2](#_Toc80091941)

[**4.1 프로젝트 수행 조직도** 2](#_Toc80091942)

[**4.2 조직별 역할** 3](#_Toc80091943)

[**5. 관리 프로세스 계획** 3](#_Toc80091944)

[**5.1 작업 계획** 3](#_Toc80091945)

[**5.2 산출물 검수 계획** 3](#_Toc80091946)

[**6. 프로젝트 관리** 3](#_Toc80091947)

[**6.1 프로젝트 세부 내용** 3](#_Toc80091948)

[**6.2 위험관리 방안** 3](#_Toc80091949)

[**6.3 이슈 및 관리방법** 3](#_Toc80091950)

[**6.4 프로젝트 후기** 3](#_Toc80091951)

# **개요**

## **1.1 프로젝트 명**

- **Green Farm** 선정한 이유는 친환경 녹색 성장 동력을 지닌 스마트팜의 미래 지향적인 가치를 추구하기 위하여 대표적인 색상인 녹색을 프로젝트명으로 채택하였습니다. 농장의 푸르름과 우리 조의 미래를 상징하는 녹색을 대표로 삼고 있습니다.

## **1.2 프로젝트 기간**

- 2021년 3월 23 – 2021년 8월 13일

## **1.3 프로젝트 목적**

코로나로 인해 사람들이 가정내에서 할 수 있는 취미를 가지는 경향을 보이고 있습니다. 이런 흐름에 편승하여 가정용 스마트팜 시장도 떠오르고 있습니다. 인터넷에서 스마트팜을 구매하려면 적게는 5만원에서 많게는 60만원까지 다양한 가격대로 형성되어 있습니다. 주로 판매되고 있는 DIY세트는 저렴한 것은 퀄리티가 너무 떨어 지고 퀄리티가 좋은 제품들은 가격대가 너무 높게 형성되어 있습니다. 보통 아두이노를 통하여 제작하지만 아두이노는 배선 체결 안정성이 떨어지고 필요하지 않은 부품도 있어 효율이 떨어집니다. 손이 덜가는 장점도 있지만 코어한 고객들은 납땜 및 회로 구성을 하는 것도 원하기 때문에 적절한 가격과 높은 퀄리티의 가성비 높은 스마트팜을 제작하는 것을 목표로 삼고 개발을 시작하였습니다.

## **1.4 프로젝트 관련 국내외 동향**

- 국내의 스마트팜 동향

농가 인구 감소 및 고령화로 인해 농산물의 생산성 및 효율성 제고 필요성이 증대되고 있는 가운데, 국제

적인 시장 개방화에 따라 농산물 경쟁력을 확보하기 위해 농업 선화가 요구되고 있어 스마트팜의 필요성이

높아지고 있다. 특히, 최근 기후변화 심화로 농산물의 작황 및 생산량이 일정하지 못해 가격이 등락하는 문

제점이 있어 수급 안정화가 필요하기에, 기상·생육 과정에 대한 데이터 축적 및 예측 강화, 기후 조건에서

자유로운 식물공장 등의 실내형 농업이 가능한 스마트팜이 큰 주목을 받고 있다. 전 세계적으로 신성장 동력

중 하나로 인식되고 있는 만큼 정부 차원의 대규모 투자 및 인프라 확충이 이루어지고 있으며, 시장주기 상

도입기에 해당하여 기술개발을 활발히 진행하고 있다.

한국의 스마트팜의 진척도은 평균 1.5세대에 있으며, 국내 농가당 경지면적은 다른 주요시장인 미국, 캐나

다에 비해 매우 작은 수준이다. 이에 따라 정부는 스마트팜을 혁신성장 주도사업으로 선정하였으며, 2022년까지 전국에 혁신밸리 4개소(상주, 김제, 밀양, 고흥) 조성 계획을 밝혔다.

- 외국의 스마트팜 동향

세계 스마트팜 시장은 2020년 352조 4800억원 규모이며, 이후 연평균 16.4%씩 성장률을 보일것으로 예상되며 2022년에는 449조 4120억원에 달할 전망이다. 1980년대 중반 미국에서 정밀농업이 등장한 이래로 미국과 캐나다는 농업 선진화가 빠르게 이루어졌으며, 대규모 기업형 농장을 중심으로 스마트팜의 주요 시장을 차지하고 있다. 향후 인구가 빠르게 증가하고 있는 인도, 중국, 동남아시아가 속한 아시아태평양 지역이 가장 높은 성장률을 보일 것으로 전망된다. 현재 네덜란드는 농식품 클러스터 구축으로 농업 첨단기술 및 시설원예 분야를 선도하고 있으며, 독일은 농업의 디지털화와 ‘디지털마을 프로젝트를 추진, 일본과 미국은 빅데이터를 활용하여 스마트농업 활성화를 추진하고 있습니다.

# **2. 프로젝트 범위**

## **2.1 작업명세서**

- smart farm 모듈 개발

|  |  |
| --- | --- |
| 업무 | 업무 범위 |
| Smart farm 모듈 개발 | - atmega128 모든 센서통합  - Arduino Uno TFT 2.4in LCD 제어  - RaspberryPi4 Server, DB |
| 문서화 | - Gantt chart, 자재명세서 |

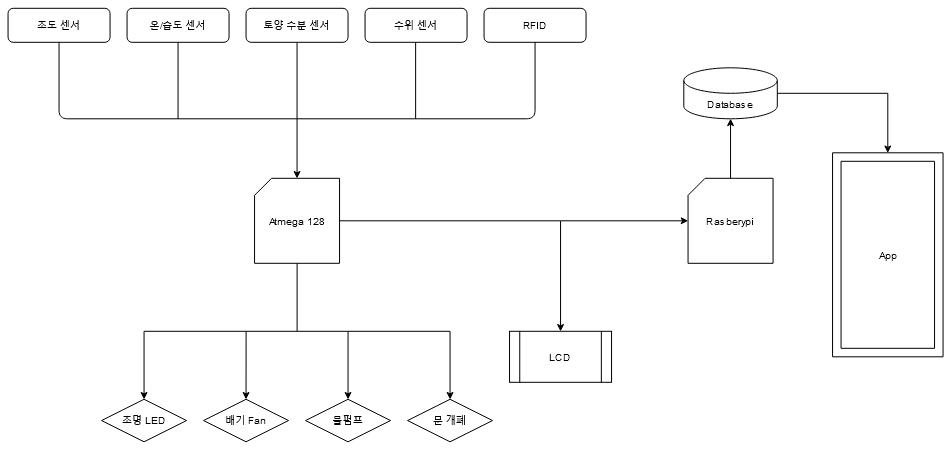
- 전체 시스템 구축

|  |  |
| --- | --- |
| 업무 | 업무 범위 |
| 시스템 구축 | Atmega128 센서의 값을 받아 환기팬, 워터펌프등의 출력 장치 제어  Atmega128의 센서 값을 Serial통신을 통하여 RaspberryPi4와 Arduino Uno에 송신  Atmega128에서 받은 센서 값을 RaspberryPi4에서 서버 구축 및 데이터베이스에  업로드, 스마트팜 본체에서 LCD로 데이터 출력  서버의 데이터베이스에 업로드된 값을 스마트폰 App을 통하여 모니터링 |
| 테스트 및 안정화 | 스마트팜 하드웨어 전체 동작 테스트 및 디버깅  RaspberryPi4를 통한 데이터베이스 App을 통해 동작 확인  LCD로 센서 출력값 실시간 확인 |

# **3. 시스템 구축 환경**

## **3.1 하드웨어 구축**

- 하드웨어 구성



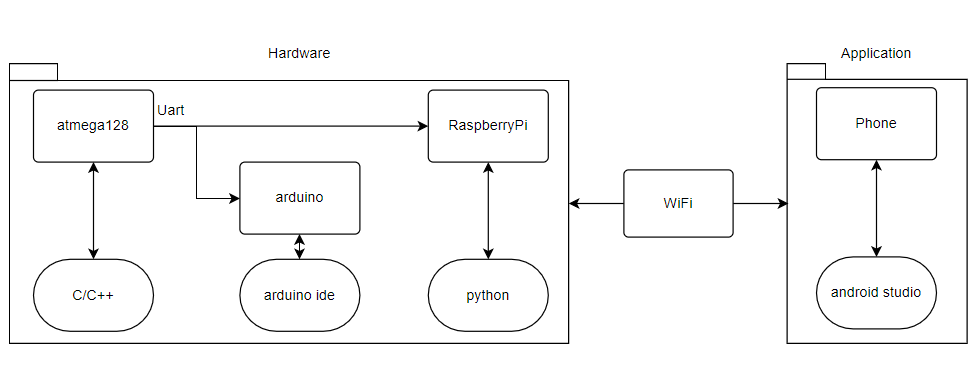
<그림 – 하드웨어 구상도>

- 하드웨어 구성 현황

|  |  |  |
| --- | --- | --- |
| 하드웨어 | 모델명 | 상세 설명 |
| Atmega128 | Atmega128 | 메인 컨트롤 MCU |
| Arduino Uno | Arduino Uno | TFT 2.4in LCD 컨트롤 B/D |
| RaspberryPi4 | RaspberryPi4 | 메인 서버, 데이터베이스 구축 |
| 온/습도 센서 | DHT11 | 온도와 습도 한 개의 센서로 감지 |
| 토양수분 센서 | SEN0193 | 녹 방지 제품 사용 |
| 조도 센서 | LLS05-A | 다이오드 타입의 센서 사용 |
| 수위감지 센서 | SEN030004 | 직접 접촉방식의 센서 사용 |
| RFID | MFRC522 | SPI 통신을 이용하여 묘듈과 통신 |
| 환기 FAN | MGA4005LR | 5V DC 모터 사용 |
| 워터 펌프 | SZH-GNP155 | 5V DC 모터 사용 |
| LED | 52X83C | 5pi 고휘도 LED 사용 |
| 서보모터 | MG996R | 토크9.4/11kg, 스피드0.17/0.14sec 4.8V/6V, 무게55g, 사이즈 40.6  \*19.8\*42.9mm |
| LCD | 2.4in TFT LCD | 2.4in TFT LCD 사용 |
| 레귤레이터 | Lm1117 | 5V 입력 3.3V 출력 레귤레이터 사용 |

## **3.2 소프트웨어 구축**

- 소프트웨어 구성



<그림 – 소프트웨어 구상도>

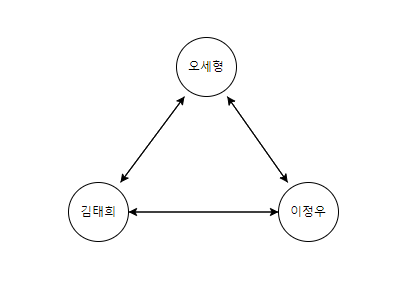
- 소프트웨어 구성 현황

|  |  |
| --- | --- |
| 운영체제 | 구성 내용 |
| Atmel  Studio7 | 센서와 통신하여 센서의 값을 받는 코드 작성  각 출력 장치를 센서의 값으로 제어  센서의 값을 Serial통신으로 각 기기에 전송 |
| Arduino ide | Atmega128과 Serial 통신 및 TFT 2.4in LCD의 제어 |
| Python3 | Atmega128과 Serial 통신 코드 작성  Serial 통신으로 받은 값을 서버를 통하여 데이터베이스에 전송 |
| MariaDB,  MySQL | 서버 구축 및 데이터베이스 작성 및 관리 |
| Android  studio | 스마트팜 모니터링을 위한 App 제작  과거의 기록을 로그로 확인 가능 |

# **4. 프로젝트 추진체계**

## **4.1 프로젝트 수행 조직도**

- 프로젝트 조직



<그림 - 조직도>

## **4.2 조직별 역할**

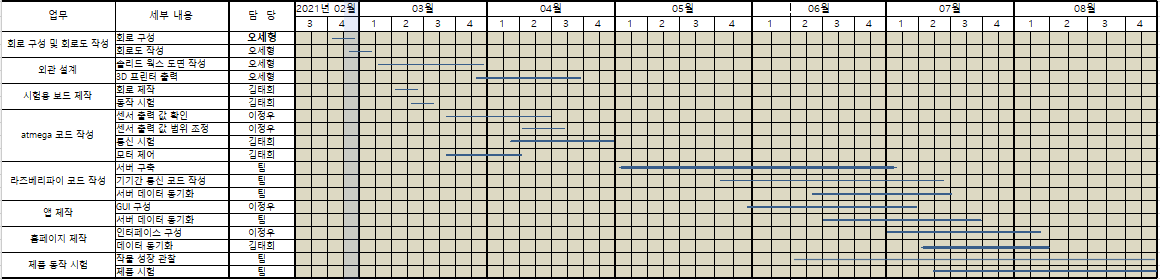
- 각 역할

|  |  |
| --- | --- |
| 성명 | 직무 |
|  | - 회로 설계, H/W 디버깅  - 배선 설계 및 정리  - 자재 관리 및 설계 데이터 관리 및 구체화 |
|  | - 순서도(Flowchart) 구조 설계  - Atmega128 제어용 Avr studio 코드 작성  - Avr studio S/W 디버깅 |
|  | - MariaDB로 서버 구축  - mysql로 데이터베이스 작성  **-** python3를 통하여 Atmega128과 Serial 통신 및 데이터베이스에 데이터 작성  - 스마트폰 App 제작 |

# **5. 관리 프로세스 계획**

## **5.1 작업 계획**

- Gantt chart



<차트 - Gantt chart>

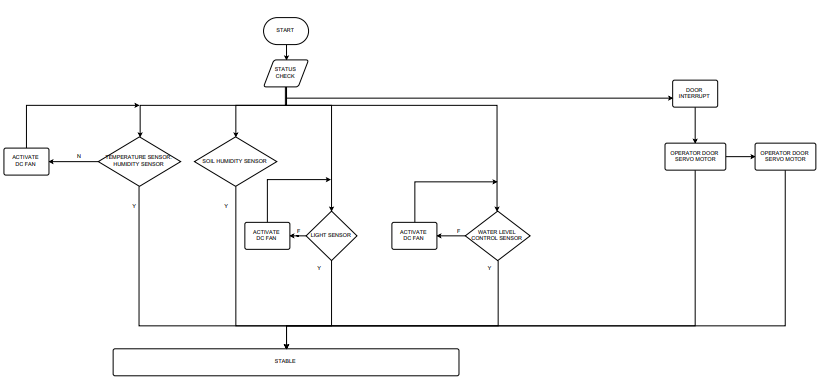
## **5.2 산출물 처리 계획**

- 재배된 무순 및 상추는 공급하는 물에 이물질이 포함되어 폐기처리함.

# **6. 프로젝트 관리**

## **6.1 프로젝트 세부 내용**

- 순서도



<그림 smart farm flow chart>

- 장비 작동 시연

a. 조도센서에 따른 LED 작동

|  |  |  |
| --- | --- | --- |
| LED OFF | | LED ON |
|  |  | |

b. 온/습도 센서에 따른 FAN 작동

|  |  |  |
| --- | --- | --- |
| FAN OFF | FAN ON | |
|  | |  |

c. 토양센서에 따른 워터펌프 작동

|  |  |  |
| --- | --- | --- |
| 모터와 수위감지 센서 | 워터 펌프 동작 | |
|  | |  |

d. RFID 마스터 모드 및 서보모터 작동

|  |  |
| --- | --- |
| 문 닫음 | 문 열림 |
|  |  |

f. LCD 동작 시연

|  |  |
| --- | --- |
| LCD MENU | LCD 조도 출력 |
|  |  |
| LCD 토양센서 출력 | LCD 온/ 습도 출력 |
|  |  |
| LCD 수위센서 출력 |
|  |

- 스마트팜의 식물재배 시연

|  |  |
| --- | --- |
| 초기 화면 | 발아 |
|  |  |
| 중기 | 생장 완료 |
|  |  |

- 어플리케이션 작동 시연

|  |  |
| --- | --- |
| 초기화면 | 동작 로그 |
|  |  |

## **6.2 위험관리 방안**

- 주의사항

1) 전원 관련 주의 사항

- 워터펌프의 전원 어댑터는 다른 전원과 같은 소스에 연결하지 말 것

- 기기의 특성상 여러 전원 소스를 사용하기 때문에 전원 체결시에 합선 주의

- 서보모터의 전류 필요량이 권장 3A이상이기 때문에 어댑터의 선정에 주의할 것

2) 문 개폐시에 주의 사항

- 제품에 사용된 서보모터는 출력이 높은 제품이기 때문에 문 동작시에 절대 손을 넣지 말 것

- 서보모터 동작시에 강제로 동작을 방해하면 스톨토크가 걸려 모터가 소손될 가능성이 있어

주의 요망

## **6.3 이슈 및 관리방법**

- 이슈

1) RFID 센서 동작 정지

-> RFID 센서가 통신 과정에서 먹통이 되는 상황이 발생

-관리방법

-> RFID 센서 리셋시에 정상 작동함

2) Serial 통신 중 프로그램 다운로드 안됨

-> 다운로드에 RX/TX 핀을 사용함

-관리방법

-> Serial통신 배선을 제거한 뒤에 다운로드 진행

## **6.4 프로젝트 후기**

- 문제점 및 개선 방안

a. 문제점

1) 워터펌프에서 윤활유가 공급용 물로 녹아 나옴

2) 수위감지 센서가 녹슬어 공급용수에 묻어 나옴

b. 개선방안

1) 워터펌프 윤활유

- 구매한 워터펌프의 가격이 저렴하여 수밀성이 떨어져서 생긴 문제로 보임

- 워터펌프를 교체하면 개선될 것으로 보임

2) 수위감지 센서

- 녹방지 처리가 되어 있지 않은 단기간용 제품을 구매함

- 장시간 전용 제품으로 교체하면 개선될 것으로 보임

- 프로젝트 코드 및 데이터시트, 참고 자료

a. 메인 코드

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#include "ADC/adc.h"

#include "DHT/DHT.h"

#include "USART/USART.h"

#include "RFID/spi.h"

#include "RFID/mfrc522.h"

#include "FUNC/FUNC.h"

#include "Sens/Sens.h"

int main(void)

{

unsigned int tem=0, humid=0, soil=0, water=0, cds\_res=0, cds\_new = 0, cds\_old=0; // 센서 값 절대 값으로 선언

float humid\_value = 0, tem\_value = 0, soil\_value = 0, water\_value = 0, cds\_value = 0; // 센서 전송값 실수로 선언

*uint8\_t* byte;

// 초기화

spi\_init();

mfrc522\_init();

Usart\_Init();

adc\_init();

PWM\_init();

PIN\_init();

// RFID 관련 레지스터 변수 초기화, 레지스터는 RFID 헤더 파일 참조

byte = mfrc522\_read(ComIEnReg);

mfrc522\_write(ComIEnReg,byte|0x20);

byte = mfrc522\_read(DivIEnReg);

mfrc522\_write(DivIEnReg,byte|0x80);

//문 닫음

PORTA |= (1<<PORTA3);

*\_delay\_ms*(20);

close;

*\_delay\_ms*(1200);

PORTA &= ~(1<<PORTA3);

while(1)

{

// 온습도 센서

humid = get\_HUM(); // 습도 데이터 입력

humid\_value = humid;

tem = get\_TEM(); // 온도 데이터 입력

tem\_value = tem;

//환풍기 조정

Venti\_con(humid);

// 토양 센서

soil = adc\_read(SOIL); // 토양센서 ADC 2번 입력

soil\_value = soil;

//물펌프

W\_Pump(soil);

// 조도 센서

cds\_new = adc\_read(CDS); // 조도 센서 ADC 1번 입력

cds\_res = (cds\_old \* 0.9) + (cds\_new \* 0.1);

cds\_old = cds\_res;

cds\_value = cds\_res;

//LED 조정

Led\_Con(cds\_res);

//물높이 센서

water = adc\_read(WATER); //수위 센서 ADC 0번 입력

water\_value = water;

// uart로 데이터값 전송

// 각 데이터의 머릿글자를 메이터 식별 및 통신 시작 신호로 사용하기 위하여 전송

// / 를 각 데이터의 끝을 알리기 위하여 전송

// 조도센서

Usart\_TX('-');

Data\_send(cds\_value);

Usart\_TX('/');

// 토양센서

Data\_send(soil\_value);

Usart\_TX('/');

//온도

Data\_send(tem\_value);

Usart\_TX('/');

//습도

Data\_send(humid\_value);

Usart\_TX('/');

//수위

Data\_send(water\_value);

Usart\_TX('\_');

RFID(); // RFID 동작 함수

}

return 0;

}

b. 데이터시트 및 전체 코드

https://github.com/ohhsehyung/Smart\_Farm

c. 참고자료

허경용 저(2014) 아두이노 상상을 스케치하다

윤성우 저(2010) 열혈 C 프로그래밍

김관중 저(2015) 스마트 팜 기술 동향 및 전망

허경용 저(2015) atmega328 프로그래밍

정재곤 저(2021) do it 안드로이드 앱 프로그래밍