

스마트 그늘막 프로젝트

설영현 | 신승엽

목차

1. 개요
2. 시스템 구성
3. 시스템 개발
4. 시연 영상 및 사진
5. 문제점 및 해결방안

기존 그늘막의 문제점



수동 조작의 불편함

기존 수동식 그늘막은 우산형 형태의 도르레로 되어있어 직접적인 인력이 투입 즉, 수동 조작으로 인한 불편함이 있습니다.



날씨 변화 대응 부족

수동 조작의 불편함 때문에 갑작스러운 기상 변화에 즉각 대응하기 어렵습니다.

스마트 그늘막의 기능 및 효과

자동화 시스템

센서 기반 자동 개폐로 사용자 편의성이 크게 향상되었습니다.

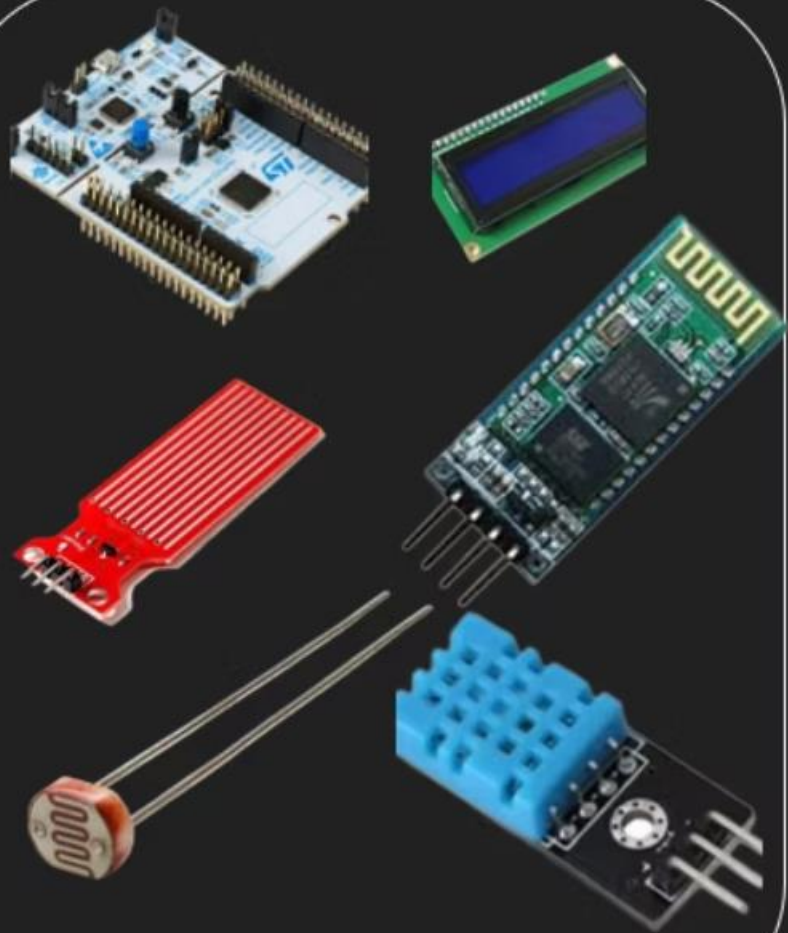
필요할 때만 작동하여 에너지 효율성이 증가했습니다.

원격 제어

모바일 앱을 통한 원격 조작이 가능합니다.

실시간 모니터링으로 상태 확인 및 즉각적인 대응이 가능합니다.

하드웨어 구성



센서
Arduino UNO R3



서버
Raspberry Pi 4B

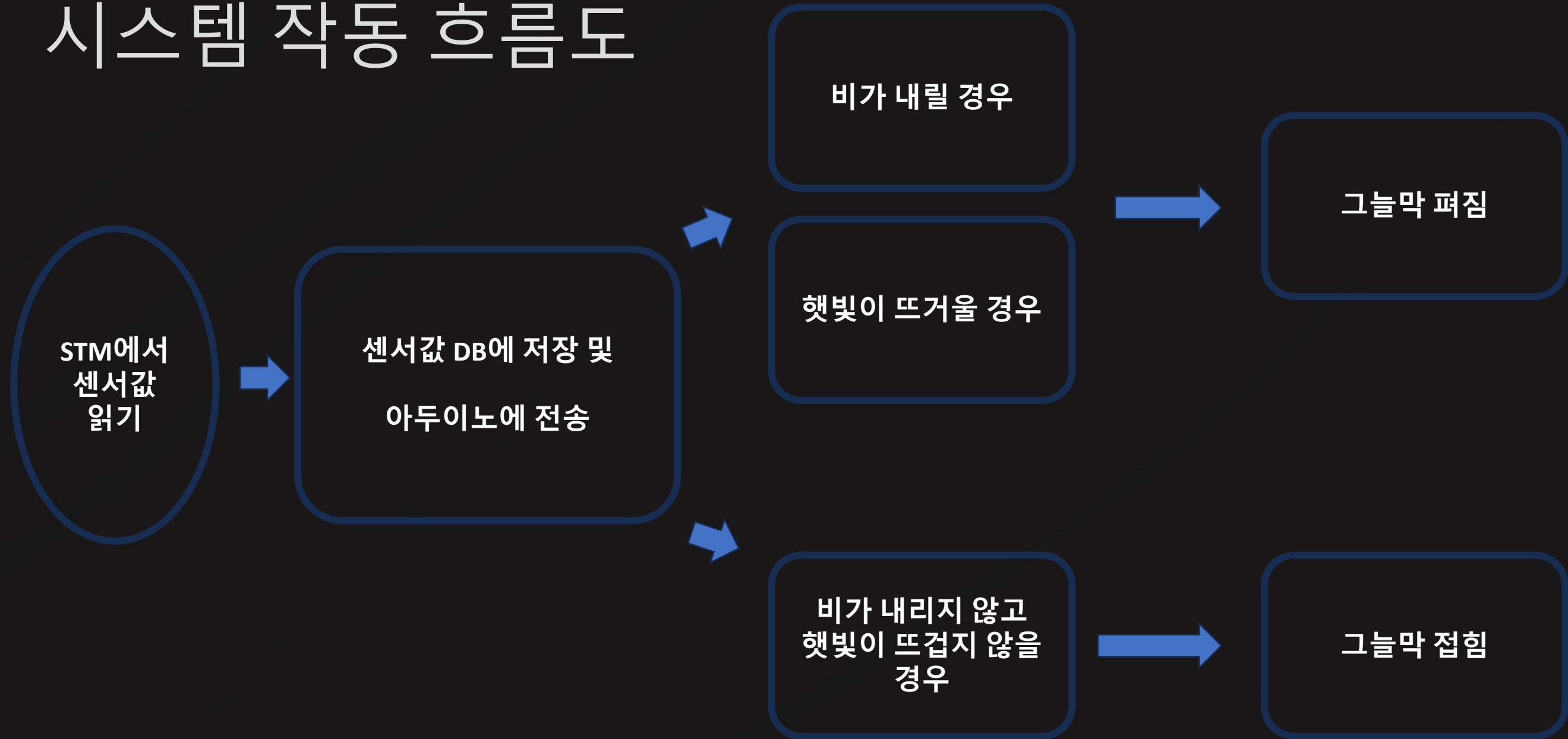


그늘막 제어
NUCLEO-F411RE

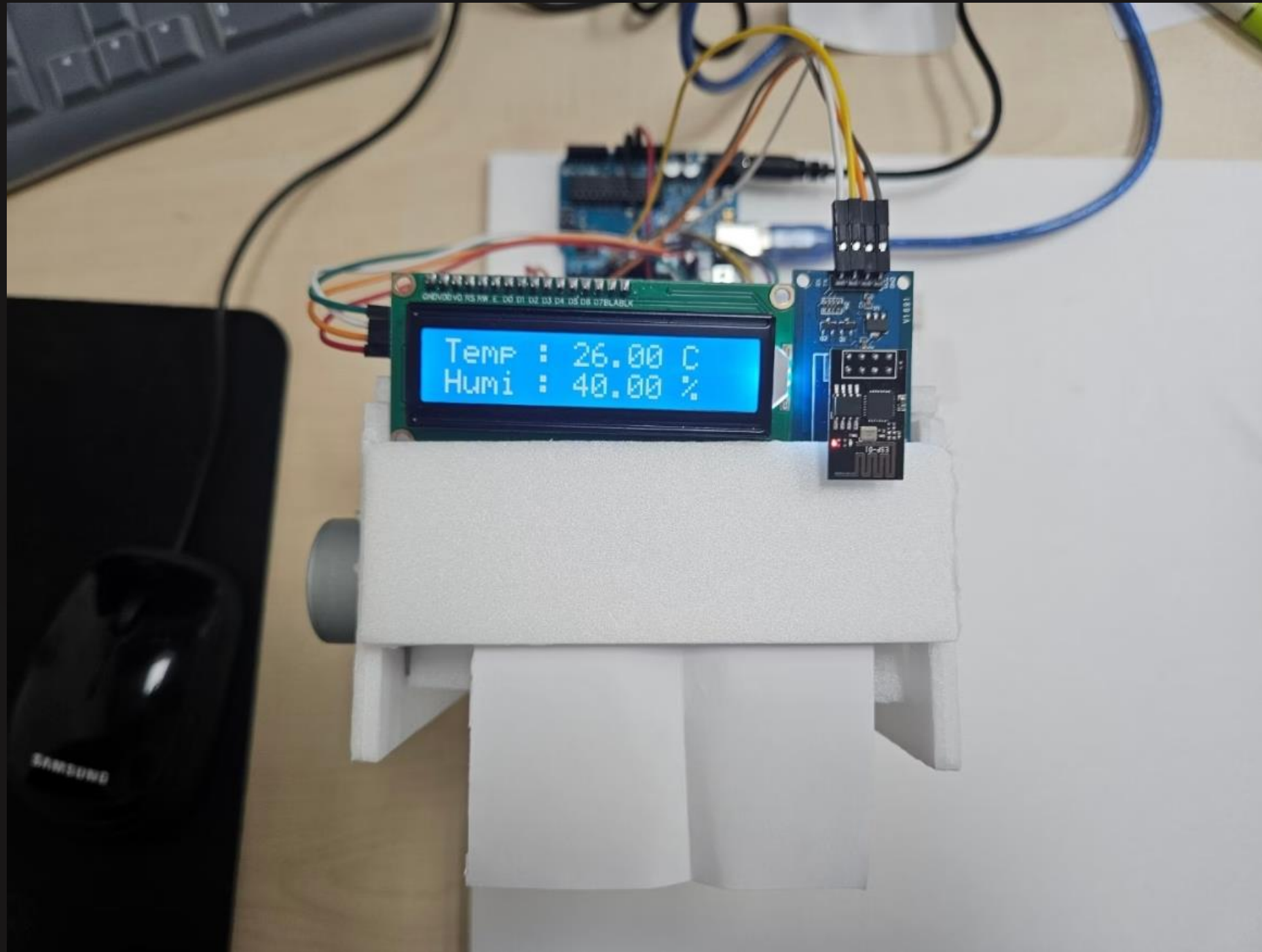
시스템 구성



시스템 작동 흐름도



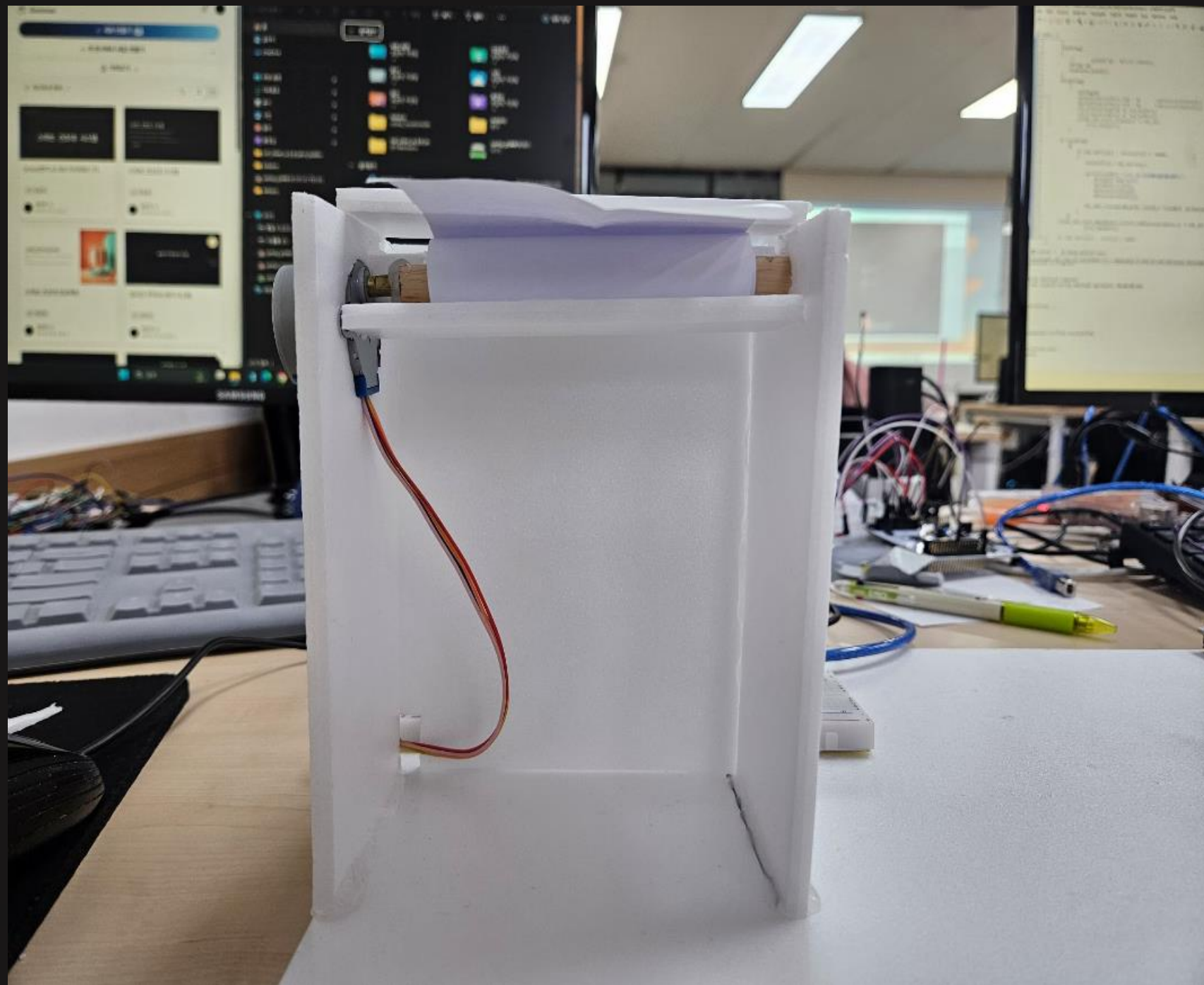
H/W 구현



아두이노 연결 모듈

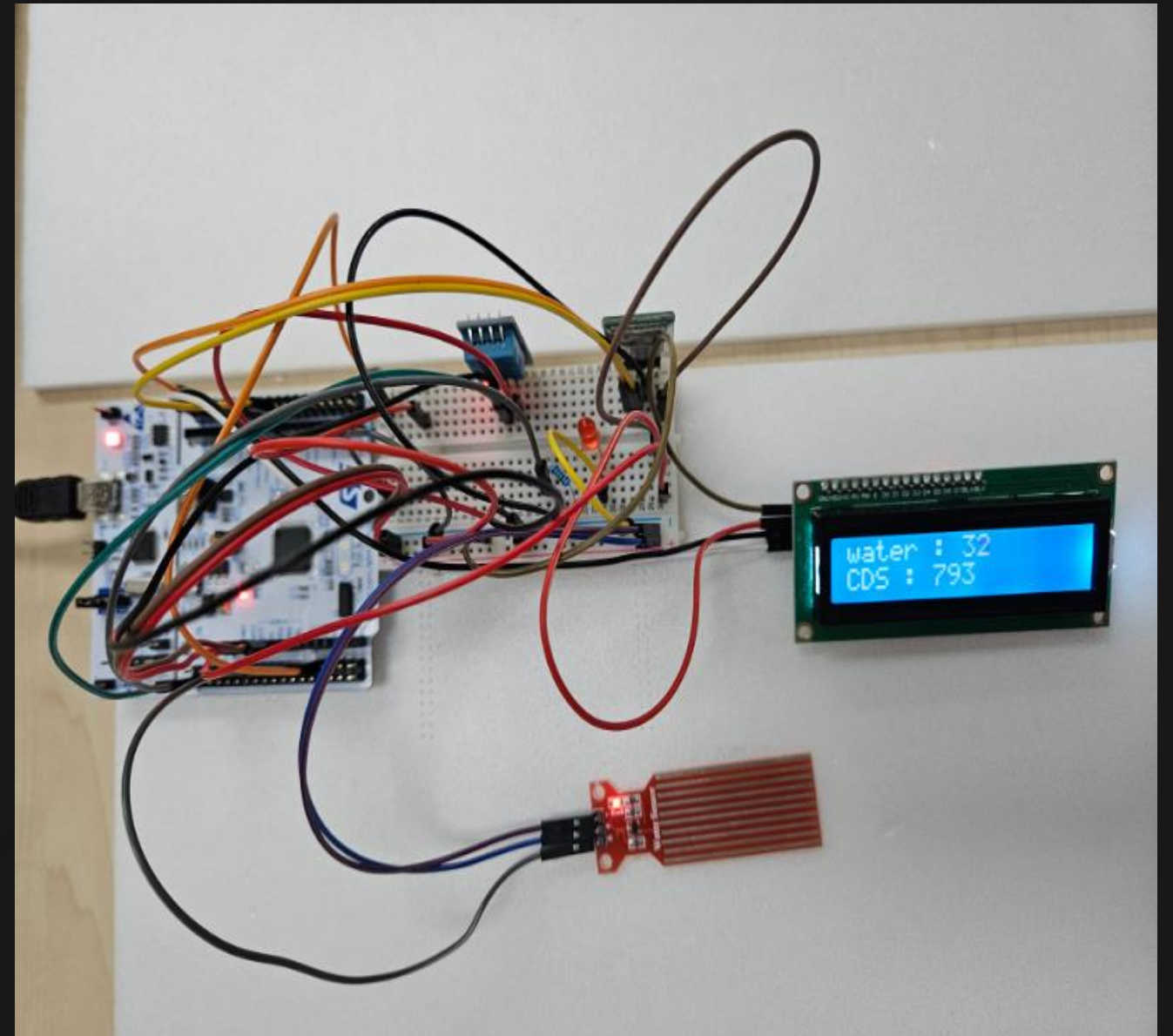
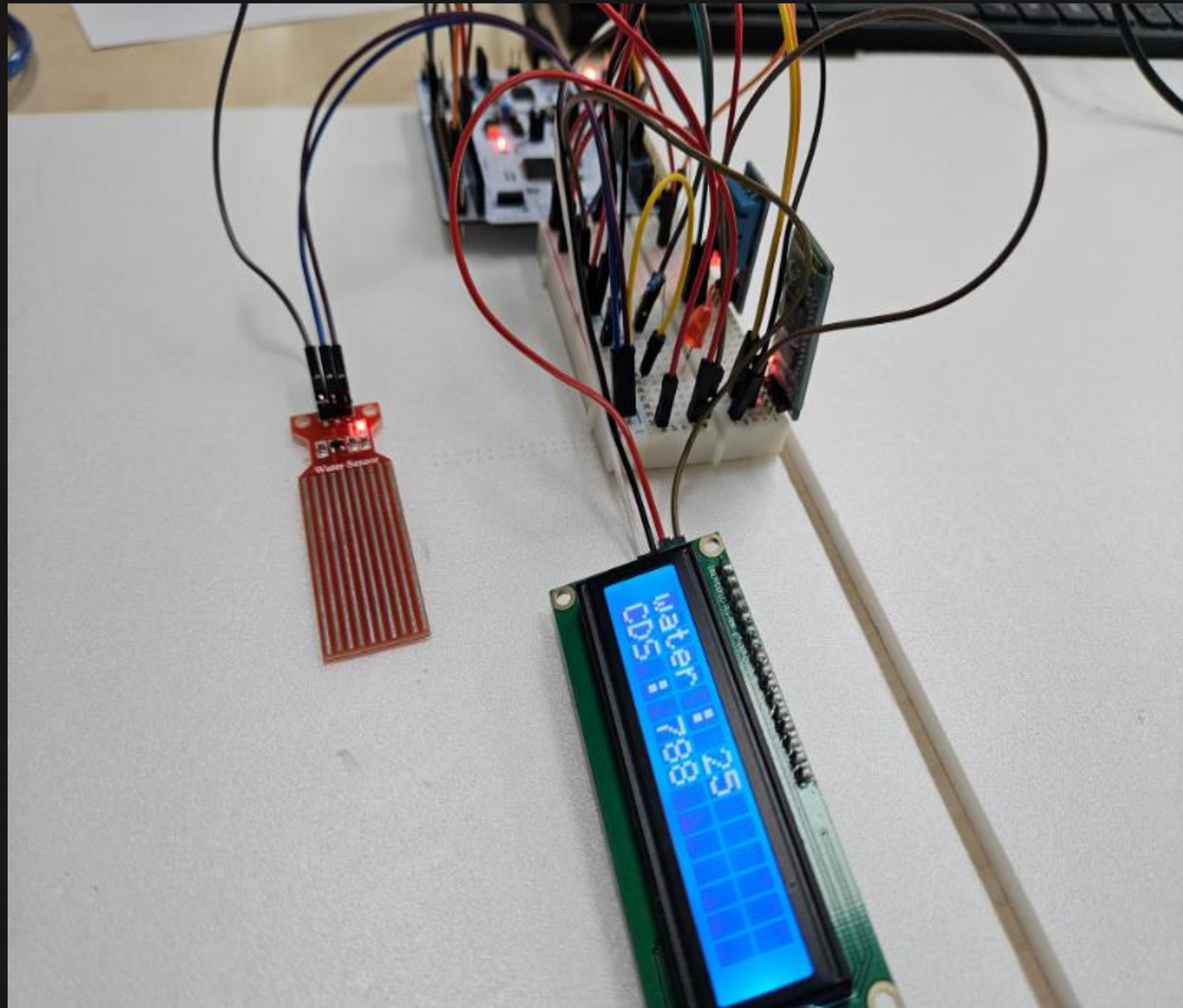
1. LCD
2. WIFI
3. STEPPER 모터

H/W 구현



H/W 구현

stm-(LCD, WATER, DHT11, CDS, BLT-HC06) 연결 사진



stm 센서 블루투스 통신

```
while (1)
{
    dht11Data = DHT11_readData();
    static uint32_t lastPrintTick = 0;
    if (HAL_GetTick() - lastPrintTick >= 1000)
    {
        lastPrintTick = HAL_GetTick();
        sprintf(led_buffer1, "water : %d", ADCxConvertValue[0]);
        sprintf(led_buffer2, "CDS : %d", ADCxConvertValue[1]);
        sprintf(buff, "h: %d%% t: %d.%d'C", dht11Data.rh_byte1, dht11Data.temp_byte1, dht11Data.temp_byte2);
        printf("water=%4d, CDS=%4d\r\n", ADCxConvertValue[0], ADCxConvertValue[1]);
        printf("%s\r\n", buff);
        LCD_writeStringXY(0, 0, led_buffer1);
        LCD_writeStringXY(1, 0, led_buffer2);
    }

    if(rx2Flag)
    {
        printf("recv2 : %s\r\n", rx2Data);
        rx2Flag = 0;
    }

    if(btFlag)
    {
        btFlag = 0;
        bluetooth_Event();
    }

    if(adcFlag)
    {
        adcFlag = 0;
    }

    if (sendFlag)
    {
        if (HAL_GetTick() - lastSendTick >= 10000)
        {
            lastSendTick = HAL_GetTick();

            sprintf(sendBuf, "[KSH_SQL]SEND@%d@%d@%d@%d\n",
                dht11Data.temp_byte1,
                dht11Data.rh_byte1,
                ADCxConvertValue[0],
                ADCxConvertValue[1]);

            HAL_UART_Transmit(&huart6, (uint8_t *)sendBuf, strlen(sendBuf), 0xFFFF);
        }
    }

    if(HAL_ADC_Start_DMA(&hadc1, (uint32_t)ADCxConvertValue, 2) != HAL_OK)
        Error_Handler();
}
```


stm 센서 블루투스 통신

```
void bluetooth_Event()
{
    int i=0;
    char * pToken;
    char * pArray[ARR_CNT]={0};
    char recvBuf[CMD_SIZE]={0};
    char sendBuf[CMD_SIZE]={0};
    strcpy(recvBuf,btData);

    printf("btData : %s\r\n",btData);

    pToken = strtok(recvBuf,"[@]");
    while(pToken != NULL)
    {
        pArray[i] = pToken;
        if(++i >= ARR_CNT)
            break;
        pToken = strtok(NULL,"[@]");
    }

    if(!strcmp(pArray[1],"LED"))
    {
        if(!strcmp(pArray[2],"ON"))
        {
            MX_GPIO_LED_ON(LD2_Pin);
        }
        else if(!strcmp(pArray[2],"OFF"))
        {
            MX_GPIO_LED_OFF(LD2_Pin);
        }
        sprintf(sendBuf,"[%s]%s@%s\n",pArray[0],pArray[1],pArray[2]);
    }
    // else if(!strcmp(pArray[1],"LAMP")){
    else if (!strcmp(pArray[1], "SEND"))
    {
        //uint32_t prevTick = 0;
        sendFlag = 1; // SEND 명령 플래그 ON
        lastSendTick = HAL_GetTick(); // 전송 타이머 초기화
        printf("%s\r\n",sendBuf);
        //송신이 안됨
    }
}
```

SQL -> DB와 아두이노에 값 전송

```
else if (!strcmp(pArray[1], "SEND") && i == 6)
{
    //printf("1\n");
    illu = atoi(pArray[2]);
    temp = atof(pArray[3]);
    humi = atof(pArray[4]);
    water = atoi(pArray[5]);

    //printf("2\n");
    // INSERT INTO shade 테이블에 저장
    sprintf(sql_cmd, "insert into shade(name, date, time, illu, temp, humi, water) values('%s', NOW(), NOW(), %d, %.1f, %.1f, %d)", pArray[0], illu, temp, humi, water);
    printf("3\n");
    res = mysql_query(conn, sql_cmd);
    if (!res)
    {
        printf("Inserted into shade: %lu rows\n", (unsigned long)mysql_affected_rows(conn));
        char response[256];
        sprintf(response, "[KSH_ARD]SEND_DATA@%.1lf@%.1lf@%d@%d\n", temp, humi, water, illu);
        //write(*sock, sql_cmd, strlen(sql_cmd));
        write(*sock, response, strlen(response));
    }
    else
        fprintf(stderr, "ERROR: %s[%d]\n", mysql_error(conn), mysql_errno(conn));
}
```

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@pi114:~\$

DB 저장된 값

phpMyAdmin

최근 즐겨찾기

information_schema
iotdb
New
device
sensor
shade
열(컬럼)
새
date (date, NULL, nullable)
humi (float, NULL, nullable)
id (PRI, int)
illu (int, NULL, nullable)
name (varchar, NULL, nullable)
temp (float, NULL, nullable)
time (time, NULL, nullable)
water (int, NULL, nullable)
인덱스
st_info

서버: localhost:3306 > 데이터베이스: iotdb > 테이블: shade

보기 구조 SQL 검색 삽입 내보내기 가져오기 테이블 작업 트리거

프로파일링 [인라인 편집] [수정] [SQL 해석] [PHP 코드 보기] [다시 보기]

<< < 79 > >> 행 갯수: 25 행 필터링: 현재 테이블 검색 Sort by key: 없음

Extra options

					id	name	date	time	illu	temp	humi	water
<input type="checkbox"/>	수정	복사	삭제		1951	KSH_STM	2025-06-05	10:11:33	26	38	32	830
<input type="checkbox"/>	수정	복사	삭제		1952	KSH_STM	2025-06-05	10:11:44	26	38	26	805
<input type="checkbox"/>	수정	복사	삭제		1953	KSH_STM	2025-06-05	10:11:55	26	38	26	772
<input type="checkbox"/>	수정	복사	삭제		1954	KSH_STM	2025-06-05	10:12:06	26	38	25	824
<input type="checkbox"/>	수정	복사	삭제		1955	KSH_STM	2025-06-05	10:12:17	26	38	28	808
<input type="checkbox"/>	수정	복사	삭제		1956	KSH_STM	2025-06-05	10:12:27	26	38	29	820
<input type="checkbox"/>	수정	복사	삭제		1957	KSH_STM	2025-06-05	10:12:38	26	38	1503	796
<input type="checkbox"/>	수정	복사	삭제		1958	KSH_STM	2025-06-05	10:12:49	26	38	1569	789
<input type="checkbox"/>	수정	복사	삭제		1959	KSH_STM	2025-06-05	10:13:01	26	38	1603	1936
<input type="checkbox"/>	수정	복사	삭제		1960	KSH_STM	2025-06-05	10:13:11	26	38	28	829
<input type="checkbox"/>	수정	복사	삭제		1961	KSH_STM	2025-06-05	10:13:22	26	38	33	805
<input type="checkbox"/>	수정	복사	삭제		1962	KSH_STM	2025-06-05	10:13:33	26	38	25	817
<input type="checkbox"/>	수정	복사	삭제		1963	KSH_STM	2025-06-05	10:13:44	26	38	29	790
<input type="checkbox"/>	수정	복사	삭제		1964	KSH_STM	2025-06-05	10:13:55	26	38	17	832
<input type="checkbox"/>	수정	복사	삭제		1965	KSH_STM	2025-06-05	10:14:05	26	38	27	807
<input type="checkbox"/>	수정	복사	삭제		1966	KSH_STM	2025-06-05	10:14:16	26	38	26	759
<input type="checkbox"/>	수정	복사	삭제		1967	KSH_STM	2025-06-05	10:14:27	26	38	33	780
<input type="checkbox"/>	수정	복사	삭제		1968	KSH_STM	2025-06-05	10:14:38	26	38	29	764
<input type="checkbox"/>	수정	복사	삭제		1969	KSH_STM	2025-06-05	10:14:49	26	38	20	756
<input type="checkbox"/>	수정	복사	삭제		1970	KSH_STM	2025-06-05	10:14:59	26	38	32	754
<input type="checkbox"/>	수정	복사	삭제		1971	KSH_STM	2025-06-05	10:15:10	26	38	26	758
<input type="checkbox"/>	수정	복사	삭제		1972	KSH_STM	2025-06-05	10:15:21	26	38	24	786
<input type="checkbox"/>	수정	복사	삭제		1973	KSH_STM	2025-06-05	10:15:32	26	38	22	753
<input type="checkbox"/>	수정	복사	삭제		1974	KSH_STM	2025-06-05	10:15:42	26	38	23	773
<input type="checkbox"/>	수정	복사	삭제		1975	KSH_STM	2025-06-05	10:15:53	26	38	25	768

스텝 모터 제어 로직

```
#include <Stepper.h>

#define STEPS 2048
Stepper stepper(STEPS, 8, 10, 9, 11); //핀 설정

int motorState = 0; //현재 모터 상태

stepper.setSpeed(12); //스텝퍼 모터 회전 속도 지정
```


스텝 모터 제어 로직

```
} else if (!strcmp(pArray[1], "SEND_DATA")) {  
    humi = atof(pArray[2]);  
    water = atof(pArray[3]);  
    cds = atoi(pArray[4]);  
    temp = atoi(pArray[5]);  
    Serial.print("온도: ");  
    Serial.println(temp);  
    Serial.print("습도: ");  
    Serial.println(humi);  
    Serial.print("수분: ");  
    Serial.println(water);  
    Serial.print("조도: ");  
    Serial.println(cds);  
}
```

```
bool isExceeded = (water > 1000 || cds > 2000);  
  
if (motorState == 0 && isExceeded) {  
    // 초기 상태에서 초과값 -> 정방향 2바퀴  
    stepper.step(STEPS);  
    motorState = 1;  
    Serial.print("turn1");  
}  
else if (motorState == 1 && !isExceeded) {  
    // 정방향 돌고 난 후, 미만값 -> 역방향 2바퀴  
    stepper.step(-STEPS);  
    motorState = 0;  
    Serial.print("turn2");  
}  
} else  
    return;
```


s/w 영상

pi@pi14: ~/iot_socket/iot_server

pi@pi14:~/iot_socket/iot_server \$./iot_server 5000
IoT Server Start!!
[KSH_AND] New connected! (ip:10.10.141.14,fd:4,sockcnt:1)

COM4 - IPuTTY

h: 39% t: 25.3°C
water= 20, , CDS= 926
h: 39% t: 25.3°C
water= 27, , CDS= 887
h: 39% t: 25.3°C
water= 24, , CDS= 905
h: 39% t: 25.3°C
water= 27, , CDS= 934
h: 39% t: 25.3°C
water= 32, , CDS= 918
h: 39% t: 25.3°C
water= 25, , CDS= 945
h: 39% t: 25.3°C
water= 25, , CDS= 881
h: 39% t: 25.3°C
water= 21, , CDS= 957
h: 39% t: 25.3°C
water= 24, , CDS= 904
h: 39% t: 25.3°C
water= 26, , CDS= 890
h: 39% t: 25.3°C
water= 24, , CDS= 896
h: 39% t: 25.3°C
water= 26, , CDS= 886
h: 39% t: 25.3°C
water= 24, , CDS= 930
h: 39% t: 25.3°C
water= 25, , CDS= 926
h: 39% t: 25.3°C
water= 22, , CDS= 881
h: 39% t: 25.3°C

pi@pi14: ~/iot_socket/iot_client_Bluetooth

pi@pi14:~/iot_socket/iot_client_Bluetooth \$

pi@pi14: ~/iot_socket/mariadb/sql_client

pi@pi14:~/iot_socket/mariadb/sql_client \$

TCP ...

8:00

TCP Telnet Terminal

zkungfu-nifqiJ 10.10.141.74:5000

ASCII

[2025. 6. 4. 오후 7:59:16] ASCII:
[KSH_AND] New connected! (ip:10.10.141.14,fd:4,sockcnt:1)

☒ Auto Scroll

ASCII: [KSH_AND:PASSWD]

Enter ASCII Command

Send ASCII

login

lampon

lampoff

getdb

setdb

getsen...

motor50

motor 1...

send

send_sql

Btn 11

Btn 12

Btn 13

Btn 14

Btn 15

Btn 16

Btn 17

Btn 18

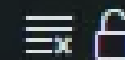
Btn 19

Btn 20

SSH is enabled and the default password for the 'pi' user has no
This is a security risk - please login as the 'pi' user and type
a new password.

S/W 영상

Output Serial Monitor




스케치는 프로그램 저장 공간 26234 바이트(81%)를 사용. 최대 32256 바이트.
전역 변수는 동적 메모리 1421바이트(69%)를 사용, 627바이트의 지역변수가 남음. 최대는 2048 바이트.



I

Uploading...

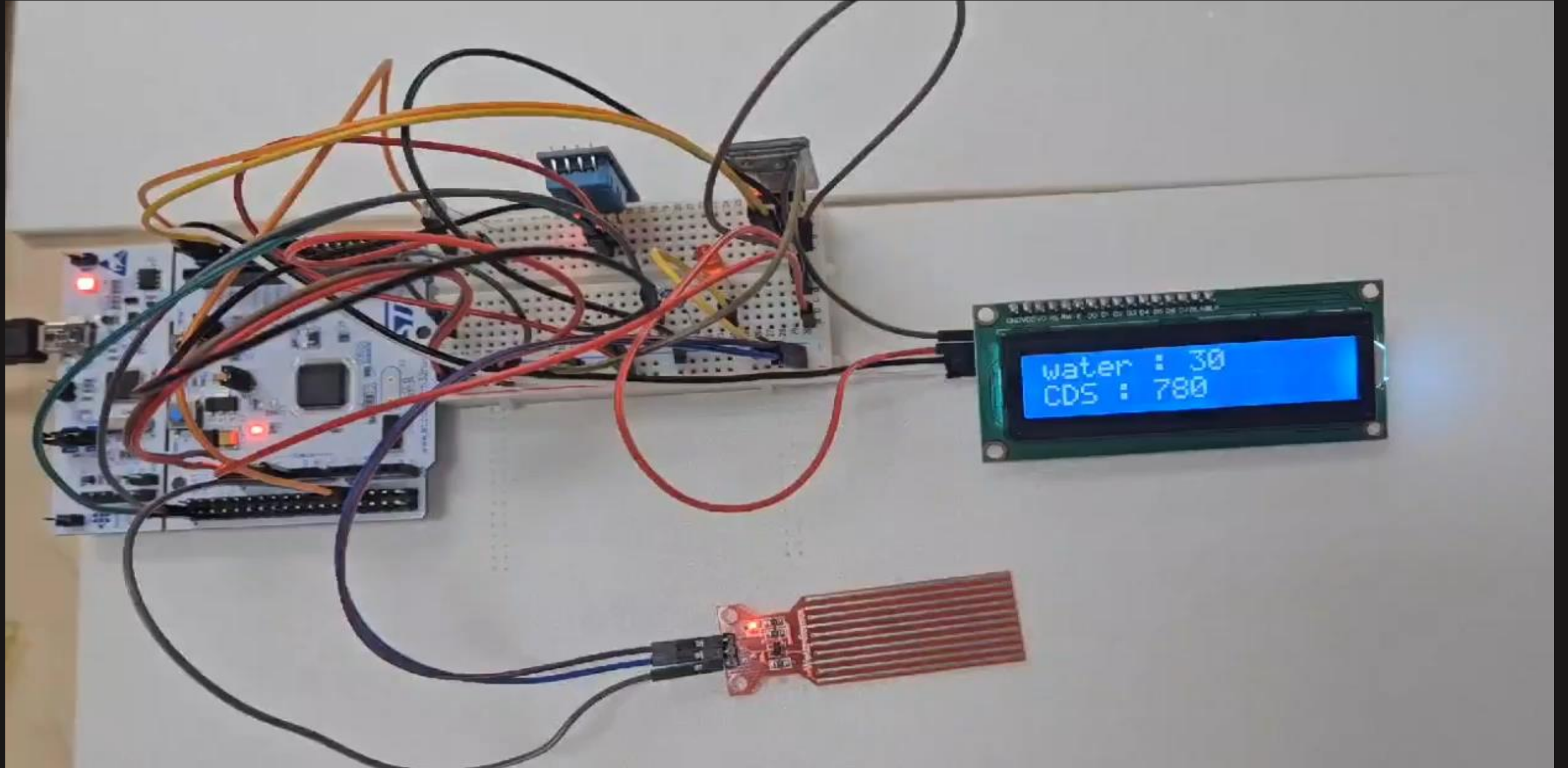
CANCEL

 Done compiling.



Ln 153, Col 3 Arduino Uno on COM3  3 

H/W 영상



H/W 영상



문제점 및 해결방안

1. 현재 그늘막의 상태값을 못 본다 : 상태값을 보여주는 코드를 추가해 비가 오거나 습도가 높을 때 그늘막이 작동하지 않으면 수동으로 작동시킨다.
2. 다양한 변수에 대응 부족 : 바람 및 지상 악화에 대한 대응을 위한 센서 및 코드 추가 필요
3. 제작하는 재료의 한계 : 보다 다양한 재료 및 아이디어로 추가 설치 필요
4. 다양한 기능 부족

프로젝트 후 느낀점

설영현

많이 미흡하지만 프로젝트를 완성했다는 점에서 뿌듯함도 느낍니다
하지만 평소 강의에서 배웠던 예제들로 프로젝트를 진행했지만 그 예제 코드들 제대로 이해하지 못 한다면 프로젝트에서도 효율적으로 사용하지 못 한다는 것을 배웠습니다. 생각보다 추가하고 싶은 기능들도 많이 생각나 재밌는 첫 프로젝트였습니다.

신승엽

처음 생각했던 거랑 다른 주제를 잡아서 예상하지 못했던 문제들이 많아서 많이 당황했습니다.
처음 배우는 통신에 많은 어려움을 겪었으나 기존에 했던 내용들을 하나씩 분석하면서 개발 중 나타난 문제들을 해결하였습니다.
생각했던 것과 달리 진행이 매끄럽진 않았으나 많은 것을 배웠습니다.

감사합니다