PJ : Cold Brew Machine

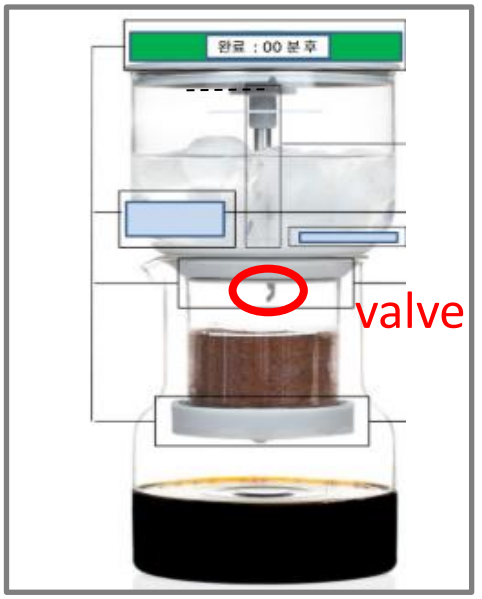
조원 : 홍기화

[프로젝트 목적] Cold Brew Coffee Automation System

주 목적

<---- optional

<---- mandatory



1 Auto Valve Control <---- Coffee Ground Analysis

Image processing

How 1 : Solenoid Valve	How 2 : Valve with Motor
on/off control (<100ms)	Motor – Water drop : P control
water drop volume calculation at different height	Water drop volume calculation at different height

2 Estimated Time Display

Estimated Time = interval * (current water volume / water drop volume)

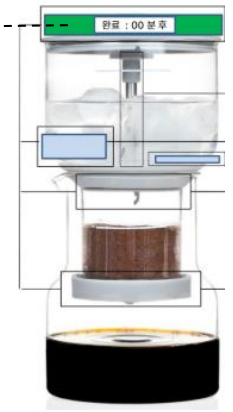
water drop volume calculation at different height

Current water volume calculation

How 1 : Distance sensor	How 2 : weight sensor

[프로젝트 구성도] Cold Brew Coffee Automation System

Done?	< Tech requirement >	< Function Block >
√	On/off control	UV sterilization
√	IR data processing	Remote Control
√	On/off control, sensing	Temperature Control
△	LCD control	Estimated Time Display
△	mcu / dsp Ethernet setup	Ethernet Communication
√	On/off motor control	Auto Valve Control
	Image Processing (Color)	Coffee Ground Analysis



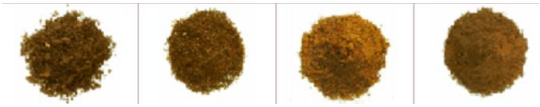
Ethernet

DSP

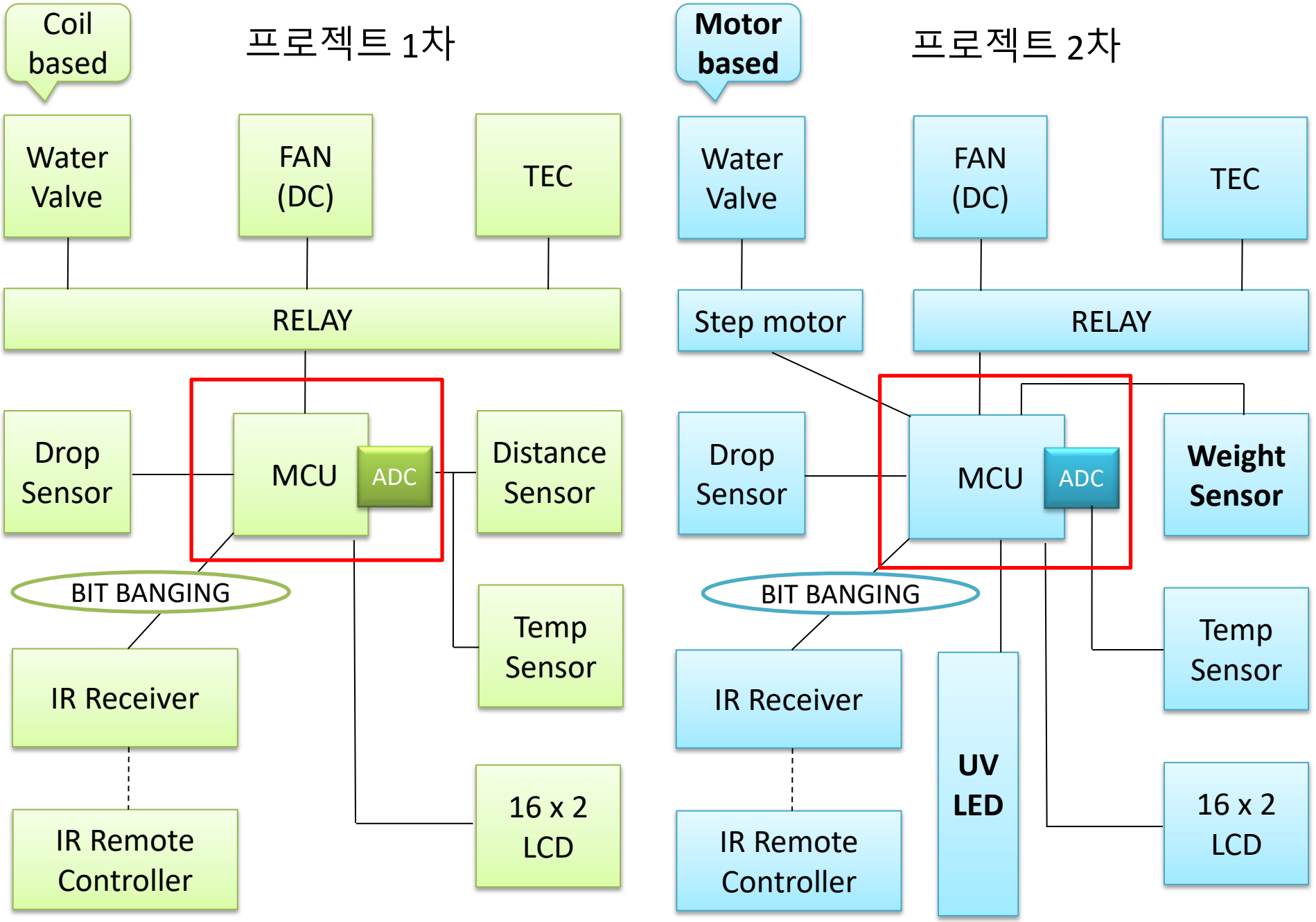
Camera

color

size



[MCU Peripheral]



[WEEK 5 : BOM]

대분류	소분류	품명	수량(ea)	단위가격(원)	총액(원)
electronic parts	MCU	TI tms570 launchpad	1	37570	37570
	LCD	LC1621 LCD	1	7700	7700
	Weight Sensor	Load cell BND-611N 1kg	1	25300	25300
	Weight sensor adc IC	HX711 Module (24bit AD)	1	1540	1540
	Drop Sensor	photo interrupter	1	1760	1760
	Temperature sensor	ETH-01DV	1	8910	8910
	UV-LED	uv-c 4545 smd led 5mW	3	5000	15000
	IR receiver + control board	(chinese)	1	2000	2000
	step motor controller	ULN2003 Module	1	1300	1300
	step motor for valve control	28BYJ-48	1	1200	1200
	FAN		1	17500	17500
	TEC Module	TEC-12705	2	5400	10800
	Relay		2	2000	4000
	Camera	usb3.0	1	70000	70000
Mechanical parts	door switch	ramps 1.4 limit switch	1	2400	2400
	펠티어 단열스폰지	펠티어 단열스폰지	1	600	600
	Peltier heatsink	Peltier-Heatsink-Set(협신전자)	1	16500	16500
	LED PCB	smd led 기판	1	1800	1800
	투명 튜브	에어호스 6mm	1	800	800
	electric wire	0.3SQ × 12C 10color 1m	1	1300	1300
	wire mold	wire duct PVC 사각몰드 밤색 1호	3	300	900
	문 경첩	경첩	2	1000	2000
	문고리	문고리	1	1000	1000
	문자석	문 자석	1	1200	1200
	case	플라베니아 5T 회색	1	4950	4950
	단열재	압축스티로폼 20mm	1	2000	2000
	물병	물병_티보틀	1	5000	5000
	실리콘마개	SL.Sto6105 (싸이랩코리아)	1	4500	4500
	Water valve	FSC0600 호스6mm 미세조절	1	2600	2600
	Water tank	daiso 1001333	2	1760	3520
	Li-po 2800mAh 35C	Li-po 2800mAh 35C	1	51900	51900
	3D Printer	Ender 3	1	230000	230000
	3D Printing material	PLA 필라멘트 1kg	1	17500	17500
	볼트너트	m3,m4,m5 볼트 너트 와셔 세트(은색)	1	6100	6100
	더치기구	더치기구	1	19500	19500
합계 총액					580650

[WEEK 5 : 상세 진행 상황, 문제 & 해결방안]

발생문제

- ❑ freeRTOS : RTI(Real Time Interrupt)를 사용할 수 없다.

-> **정밀 시간 카운트, 인터럽트 사용불가**

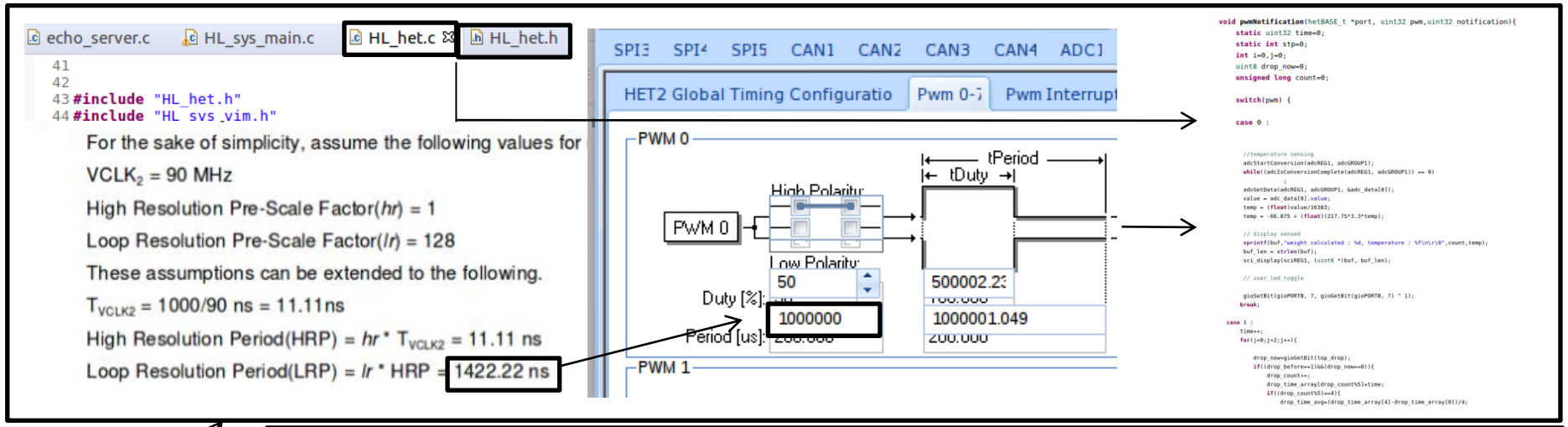
- ❑ freeRTOS에서 인터럽트 중첩시 프로그램 SKIP됨

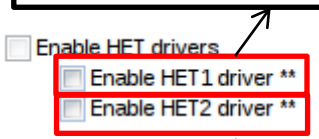
해결방안

- ❑ 원인분석 : RTI 타이머를 freeRTOS의 TICK으로 사용하기 때문

- ❑ 해결 : **1**. 보드 제공하는 HET(High End Timer) 포트 중 하나를 PWM Period End Interrupt로 사용
  -> 우선순위 높아 SKIP되지 않음

  2. 나머지 포트를 정밀 Timer 로 사용(**75MHz**)






```
void wait_66us(uint32 delay){
    delay*=5000;//100us
    hetRAM1->Instruction[0].Data=0; // hetResetTimestamp()와 상
    while(((hetRAM1->Instruction[0].Data)>>7)<delay)
        ;
}
```

```
for(;;)
{
    /* Taggle HET[1] with timer tick */
    gioSetBit(hetPORT1, 17, gioGetBit(hetPORT1, 17) ^ 1);
    gioSetBit(gioPORTB, 6, gioGetBit(gioPORTB, 6) ^ 1);
    wait_66us(6710);  // 6710*5000=33,550,000으로 2^25(33554432)
```

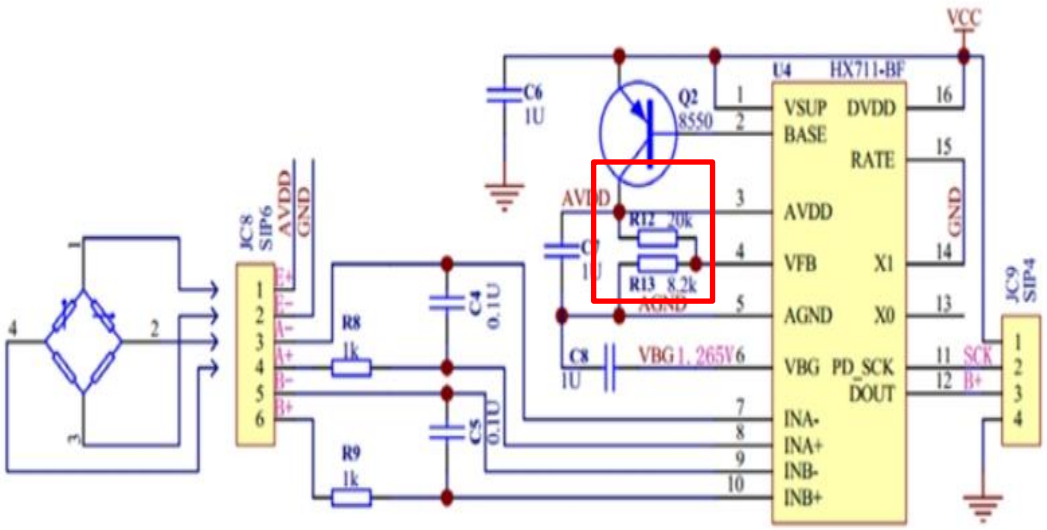
[WEEK 5 : 상세 진행 상황, 문제 & 해결방안]

발생문제

- ❑ 기본값 계속 상승 :
  전원부 캐패시터 장착
  -> 효과없음

해결방안

- ❑ 원인분석 : 5V용 보드 – 내부 레귤레이터 분압저항 수정필요.
- ❑ 해결 : R12, R13 각 20k Ohm, 8.2kOhm 에서 R12= 10 Ohm으로 수정



Loadcell ---- HX711 연결회로

해결 전 -> 후

[WEEK 5 : 상세 진행 상황, 문제 & 해결방안]

발생문제

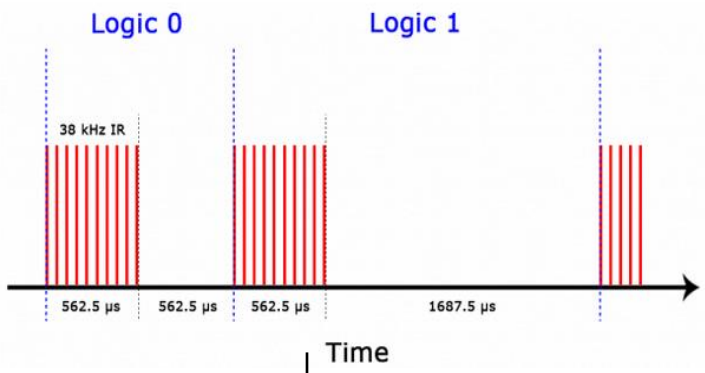
- ❑ IR 버튼 동작 불안정

해결방안

- ❑ 원인분석 : BIT BANGING 시 TIMER 부정확
- ❑ 해결 : High End Timer 사용(**13.3ns**)

13.3ns 단위로 IR CHOP PERIOD(562.5us)를
정확하게 맞춰 정확한 동작 수행

Logical '1' starts with a 562.5 µs long HIGH pulse of 38 kHz IR followed by a 1,687.5 µs long LOW pulse.
Logical '0' is transmitted with a 562.5 µs long HIGH pulse followed by a 562.5 µs long LOW pulse:

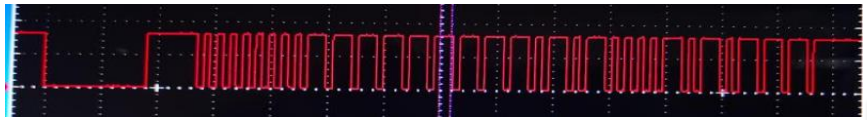


↓

신호 : 구간 분리

↓

0000 0000 0000 0000 1111 1111 01010101
01010101 0111 0111 0111 0111 0111 0111 0111 0111



```
void decode_ir(void){
    int i=0;
    //_disable_IRQ_interrupt_();

    while(gioGetBit(gioPORTA,7)==1){

    }

    for(i=0;i<122;i++)
    {
        //delay_4us(167);  //670us
        wait_us(562);

        if(gioGetBit(gioPORTA,7)==1)
            bitcount[i]=1+48;
        else
            bitcount[i]=0+48;
    }

    decode=1;

    sci_display(sciREG1,bitcount,122);
```

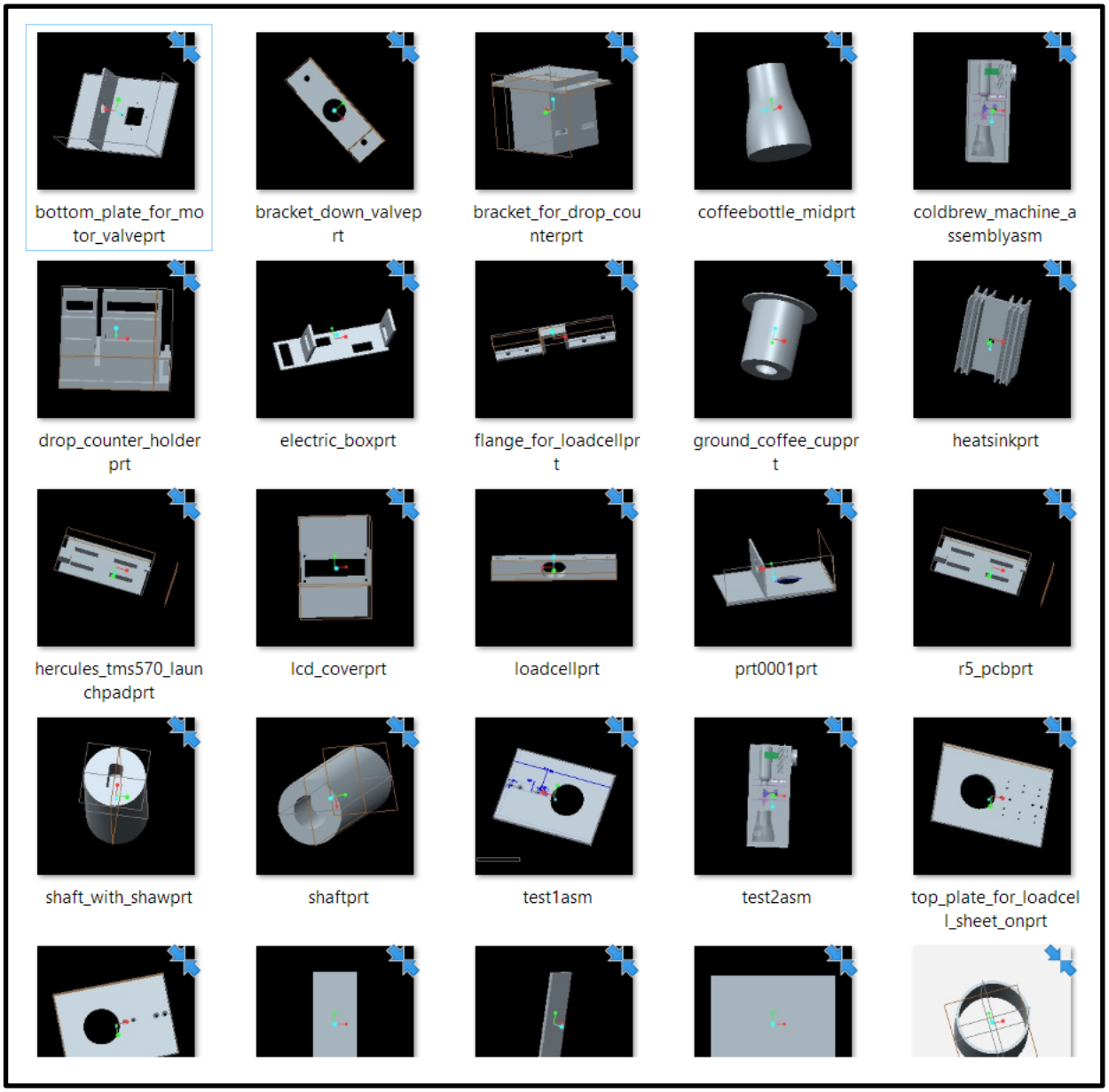
562us단위로
찢는다.

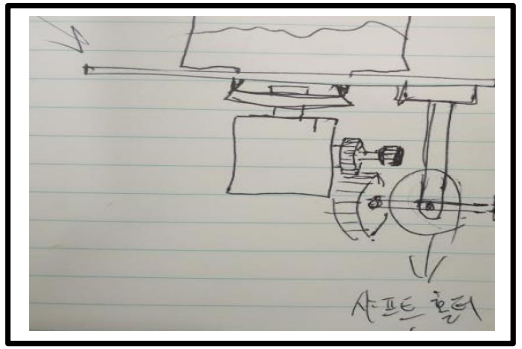
```
switch (letter[0]){
case 0xd7 :

    switch (letter[1])
    {
        case 0xd7 :

            if(letter[5]==0xd5) remote_now=7;
            else                remote_now=3;
            if(letter[2]==0xd5) remote_now=6;
        break;

        case 0xd5 :

            if(letter[2]==0xdd)
            {
                switch (letter[3])
                {
                    case 0xdd : remote_now='h';   // sharp
                    break;
                    case 0xd5 : remote_now=1;
                    break;
                    case 0xd7 : remote_now=8;
                    break;
                }
            }
            else
            {
                switch (letter[3])
                {
                    case 0xd5 : remote_now=9;
                    break;
                    case 0xd7 : remote_now=0;
                    break;
                }
            }
        break;
    }
break;

case 0xd5 :
    switch (letter[1])
    {
        case 0xdd :
            switch(letter[2])
            {   case 0xdd :
                    if(letter[3]==0xd7)
                                remote_now='s';   //s=star;
                    else        remote_now=2;
                break;
                case 0xd7 :     remote_now='r';   //r=right
                break;
                case 0xd5 :     remote_now='d';   //d=down
                break;
            }
        break;

        case 0xd7 :
            if(letter[2]==0xd7) remote_now='k';   //o=ok
            else                remote_now=4;
        break;

        case 0xd5 :
            switch (letter[3])
            {
                case 0xdd :     remote_now='u';   //u= up
                break;
```

각 버튼으로
분리

[WEEK 5 : 진행 상황]

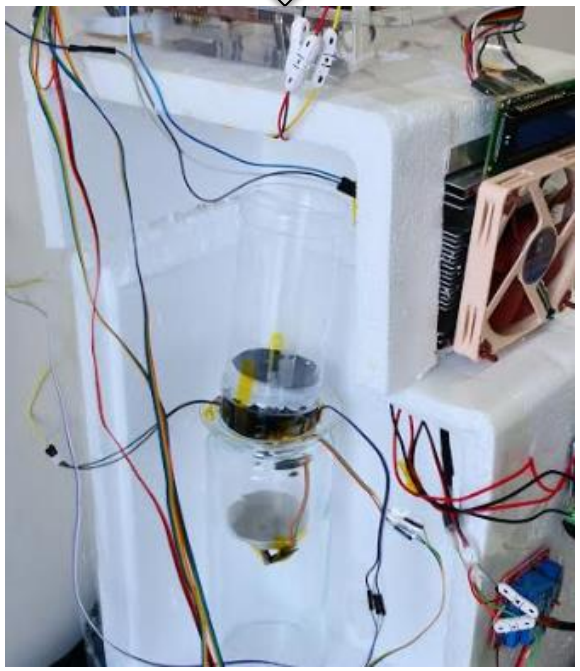
기존 모델 -> 3D 기계 설계 -> 3D 프린팅



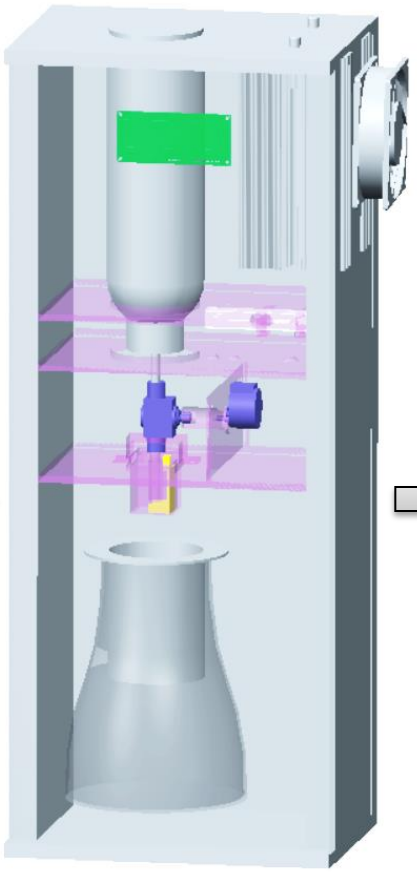


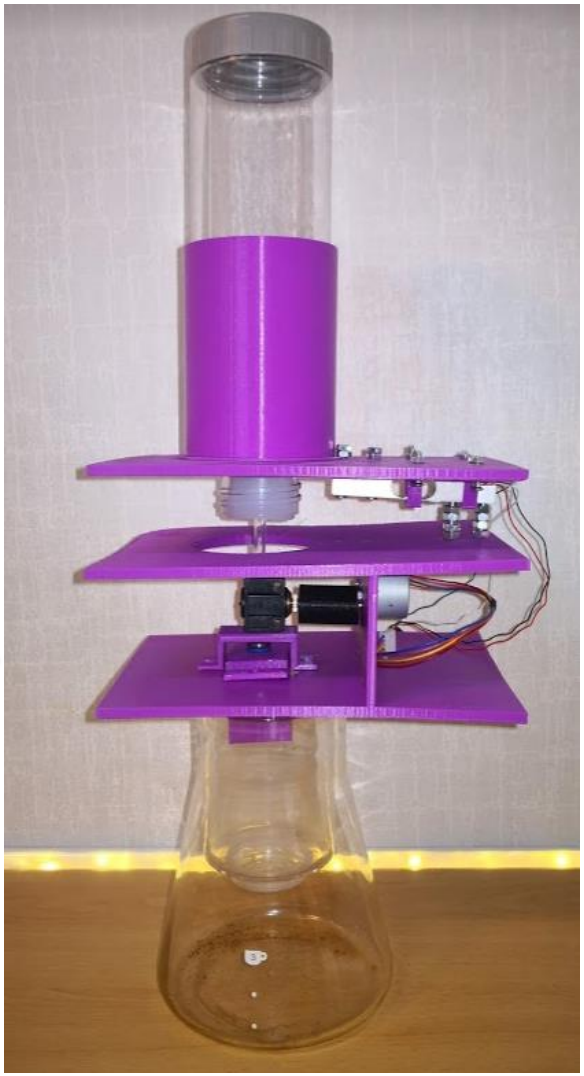
[WEEK 5 : 진행 상황]

기존 모델 -> 3D 기계 설계 -> 3D 프린팅



“ 테이핑은 그만…”





[WEEK 5 : 진행 상황]

- ❑ FreeRTOS – TCP/UDP 네트워크 내용 학습 및 Tutorial 따라 진행.
- ❑ 아래 UDP 코드 만들었으나 구동되지 않음 -> Server 측 UDP 맞게 고쳐야
  -> Client 측 task 다중 수행 부분 수정해야

echo_server.c | chat_server.c | HL_sys_main.c | HL_sys_main.c

```
    vTaskStartScheduler();

    /* Run forever */
    while(1);
/* USER CODE END */
}
static void vSendUsingStandardInterface( void *pvParameters )
{
xSocket_t xSocket;
struct freertos_sockaddr xDestinationAddress;
char cString[ 50 ];
uint32_t ulCount = 0UL;
const TickType_t x1000ms = 1000UL / portTICK_PERIOD_MS;

    /* Send strings to port 10000 on IP address 192.168.0.200. */
    xDestinationAddress.sin_addr = FreeRTOS_inet_addr( "192.168.0.195"
    xDestinationAddress.sin_port = FreeRTOS_htons( 1100 );

    /* Create the socket. */
    xSocket = FreeRTOS_socket( FREERTOS_AF_INET,
                               FREERTOS_SOCK_DGRAM,
                               FREERTOS_IPPROTO_UDP );

    /* Check the socket was created. */
    configASSERT( xSocket != FREERTOS_INVALID_SOCKET );

    /* NOTE: FreeRTOS_bind() is not called.  This will only work if
    ipconfigALLOW_SOCKET_SEND_WITHOUT_BIND is set to 1 in FreeRTOSIPCon

    for( ;; )
    {
        /* Create the string that is sent. */
        sprintf( cString,
                 "Standard send message number %lu\r\n",
                 ulCount );

        /* Send the string to the socket.  ulFlags is set to 0, so the
        semantics are used.  That means the data from cString[] is copi
        into a network buffer inside FreeRTOS_sendto(), and cString[] c
        reused as soon as FreeRTOS_sendto() has returned. */
        FreeRTOS_sendto( xSocket,
                         cString,
                         strlen( cString ),
                         0,
                         &xDestinationAddress,
                         sizeof( xDestinationAddress ) );

        ulCount++;

        /* Wait until it is time to send again. */
        vTaskDelay( x1000ms );
    }
}
```

< MCU Client >

```
#include <stdio.h>
#include <stdlib.h> // exit() 쓰기위해 필요하다.
#include <string.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <sys/socket.h>

typedef struct sockaddr_in  si;
typedef struct sockaddr *  sap;

void err_handler(char *msg)
{
        fputs(msg, stderr);
        fputs('\n',stderr);
        exit(-1);
}

int main(int argc, char **argv)
{
    int serv_sock, clnt_sock;

    si serv_addr;
    si clnt_addr;
    socklen_t clnt_addr_size;

    char msg[]="hello network programming !!\n";

    if(argc!=2)
    {
        printf("usage %s <port>\n",argv[0]);
        exit(-1);
    }
    serv_sock=socket(PF_INET,SOCK_DGRAM, 0);

    if(serv_sock == -1)
        err_handler("socket()error");

    memset(&serv_addr,0 ,sizeof(serv_addr));
    serv_addr.sin_family=AF_INET;
    serv_addr.sin_addr.s_addr = htonl(INADDR_ANY);
    serv_addr.sin_port = htons(atoi(argv[1]));      //포트번호설정
```

< DSP측 : UDP Server >

느낀점

- 회사에서 기구설계팀을 쓰는 이유가 이해되었다.
- 스스로 찾아서 할 줄 알아야 한다.
- 혼자보다는 협동하는게 더 좋다.
- 일정 마지막에는 밤을 새게 된다.