

2025 임베디드프로세서2
Term-Project 보고서

**실시간 3차원 전방 객체 탐지를 위한
3축 IMU 구형 짐벌(Gimbal) 제어**

소 속	전자공학부 임베디드시스템 전공
학 번	2021146036
성 명	최지현
제출일	2025.12.16

목 차

1. 프로젝트 필요성	02
2. 프로젝트 개요	03
3. 수행계획서	04
4. 주요 핵심 기술	05
5. 중간 보고	06
6. 수행 내용	07
7. 계획 대비 달성도 평가	09
8. 학습자 성찰	11
9. 맺음말	12

1. 프로젝트 필요성



- 현대 전쟁에서 자폭 드론 공격을 주력으로 사용하고 있다. 자폭 드론을 일반 보병이 소총으로 격추하기 어렵다. bunker, 야전 참호 등에서 비행체 추적에 적합한 짐벌 제어로 자폭 드론에 유연하고 정밀하게 대처할 수 있는 기술이 개발되어야 한다.

2. 프로젝트 개요

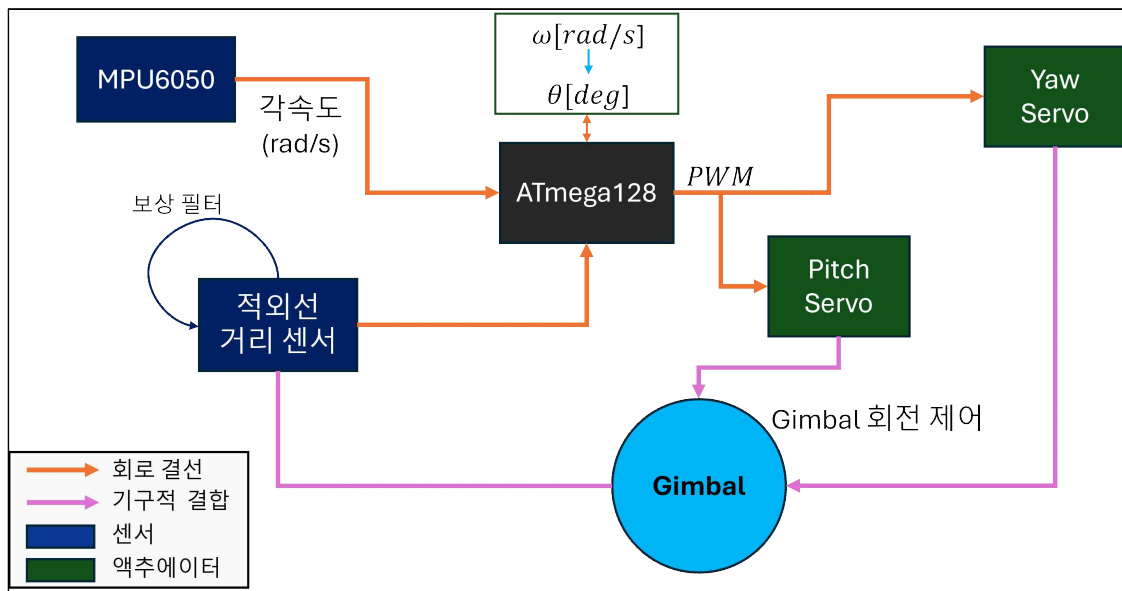
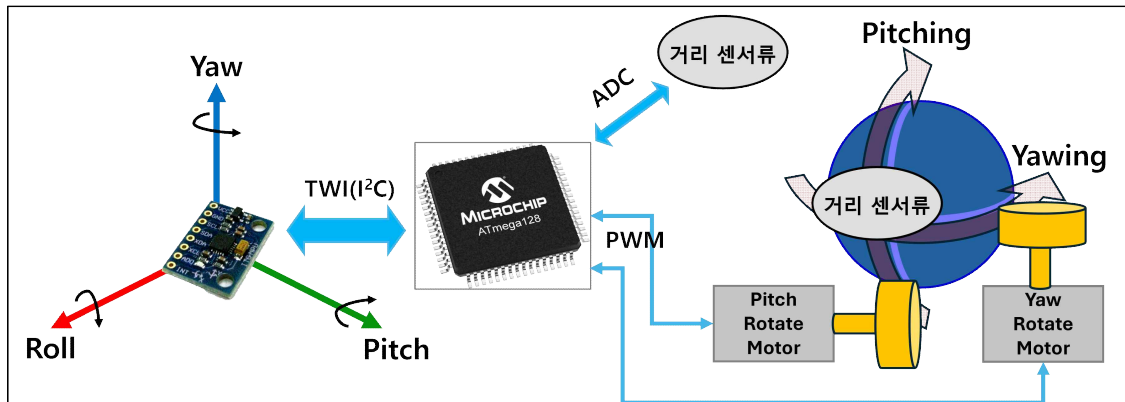
(1) 프로젝트 목표

실시간으로 구(Sphere) 정면에 위치한 거리 센서로 3차원 전방에 비행체를 탐지하도록 3축 IMU로 읽은 yaw, pitch로 구형 짐벌의 움직임을 제어하며 짐벌은 고각 0~80[deg], 방위각 -60~60[deg]의 커버리지를 갖는 시스템을 개발하고자 함.

(2) 시스템의 주요 기능

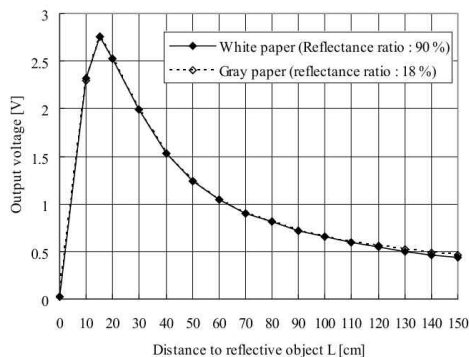
- (2-1). IMU 센서를 Pitch 방향으로 움직였을 때 짐벌도 같은 pitch 방향으로 움직인다.
- (2-2). IMU 센서를 Yaw 방향으로 움직였을 때 짐벌로 같은 Yaw 방향으로 움직인다.
- (2-3). 적외선 거리센서는 20cm~100cm까지 측정할 수 있다.
- (2-4). IMU센서가 위치한 조종기(Controller)에 LED를 부착하여 짐벌의 yaw 각도가 얼마만큼 돌고 있는지 LED 5개를 이용하여 사용자에게 보여준다.
- (2-5). 조종기에 3색 LED를 부착하여 객체가 50cm이내에 탐지 되었을 때 점등하여 객체의 유무를 파악 한다.

(3) 시스템 구성 및 프로그램 흐름 설명(구성도와 플로우차트 등)



(4) 프로젝트에 사용된 주요 구성요소

1. MPU6050: Gyroscope의 분해능을 250[deg/sec], Accelerometer의 분해능을 $\pm 4[g]$ 설정
Gyroscope의 LPF 설정, Accelerometer의 HPF 설정, Timer3로 100Hz Sampling
2. GP2Y0A02(적외선 거리센서): 출력 전압으로 거리를 측정할 수 있음



3. MG996R: Servo motor, 동작 주기 20[ms] 기준, 0[deg] -> 펄스폭 1[ms]
180[deg] -> 펄스폭 2[ms] Timer1으로 FastPWM 사용

3. 프로젝트 수행 계획서

팀별 과제수행계획서(문제 분석지)

모듈명	임베디드프로세서2		
팀 이름	4조	날짜	
팀원	최지현		

학습 목표	실시간 3차원 전방(forward) 객체 탐지를 위한 3축 IMU로 구(Sphere)형 짐벌(Gimbal) 제어
-------	--

가설/해결안 (Ideas)	알고 있는 사실 (Facts)	더 알아야 할 사항 (Learning Issues)	앞으로의 계획 (Action Plan)
플랜트를 3자유도 공간의 움직임을 직관적으로 제어하려면 플랜트가 제어기와 똑같이 3자유도 공간 움직임을 모방하면 된다.	제어기로 사용할 MPU6050(3축 IMU센서)만 사용하면 지자기 보정이 되지 않기 때문에 Yaw가 drift가 되어 움직이지 않아도 시간이 지나면 값이 바뀌어 전혀 다른 값이 나온다. GP2Y0A02(적외선 거리센서)는 거리에 따라서 전압이 바뀌는데 0~20cm에서 급격한 증가를 가지고 20cm 이후에 전압 역전 현상으로 거리가 멀어짐에 따라 전압이 다시 감소한다.	MPU6050의 Drift를 없애기 위해서 현재각도-이전각도 만큼의 각 변위량을 Servo Motor에 사용하면 적절한 Kp(비례이득계수)로 servo motor의 움직임을 제어하여 적절한 각도로 움직일 수 있도록 해야 한다. ADC의 변환 값을 고려하여 안정적으로 동작할 수 있는 센서 거리 25~ 120cm로 사용을 제한한다. servo motor를 쓰기 위해 20ms 주기의 구형파 사용	Servo motor (mg995r)이 데이터시트 상으로 duty cycle 1ms가 0°, 2ms가 180°지만 실제로 0.5ms가 0°, 2.5ms가 180°로 작동 되게 해야함. Timer의 FastPWM을 사용 MPU6050을 UART로 했을 때는 각도가 많이 튀어 TWI로 변경하고 MahonyAHRS 알고리즘으로 결측치를 성공적으로 제거해야 함. 3축 IMU의 drift문제를 해결하기 위해 주기적으로 각도를 기준으로 저장하고 기준 각도와 현재 각도의 차이만큼으로 서보모터 제어해야 함

교수 comment	
------------	--

4. 주요 핵심 기술

- 16×2 크기의 LCD 제어
- PORTC GPIO LED제어
- ServoMotor 제어
- I2C(TWI) 통신으로 Atmega128 <-> MPU6050
- MahonyAHRS PI제어 read MPU6050 알고리즘

Algorithm: MahonyAHRS

input: gx, gy, gz, ax, ay, az, Current Quaternion:(q0, q1, q2, q3)
output: Qtn_filter

Initialize: $q0 \leftarrow 1.0, q1 \leftarrow 0.0, q2 \leftarrow 0.0, q3 \leftarrow 0.0, FBx = FBy = FBz \leftarrow 0$

$Gyro \leftarrow \begin{bmatrix} gx \\ gy \\ gz \end{bmatrix}, Accel \leftarrow \begin{bmatrix} ax \\ ay \\ az \end{bmatrix}, Qtn \leftarrow \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix}, Ifb \leftarrow \begin{bmatrix} FBx \\ FBy \\ FBz \end{bmatrix} \{integral\ feedback\}$

if $ax \neq 0$ and $ay \neq 0$ and $az \neq 0$ **then**

$recipNorm \leftarrow 1 / \sqrt{ax^2 + ay^2 + az^2}$

$Accel \leftarrow Accel \times recipNorm$

$V_{half} \leftarrow \begin{bmatrix} vx \\ vy \\ vz \end{bmatrix} = \begin{bmatrix} q1 * q3 - q0 * q2 \\ q0 * q1 + q2 * q3 \\ q0 * q0 - 0.5 + q3 * q3 \end{bmatrix}$

$Error_{half} \leftarrow \begin{bmatrix} ex \\ ey \\ ez \end{bmatrix} = \begin{bmatrix} ay * vz - az * vy \\ az * vx - ax * vz \\ ax * vy - ay * vx \end{bmatrix}$

if $2K_i > 0.0$ **then**

$Ifb \leftarrow \begin{bmatrix} FBx \\ FBy \\ FBz \end{bmatrix} = \begin{bmatrix} FBx \\ FBy \\ FBz \end{bmatrix} + \begin{bmatrix} 2K_i \Delta t & 0 & 0 \\ 0 & 2K_i \Delta t & 0 \\ 0 & 0 & 2K_i \Delta t \end{bmatrix} Error_{half}$

$Gyro \leftarrow Gyro + Error_{half}$

else

$Error_{half} \leftarrow 0$

end

$Gyro \leftarrow Gyro + 2K_p \cdot V_{half}$

end

$Gyro \leftarrow 0.5 \Delta t \cdot Gyro$

$Qtn \leftarrow Qtn + \begin{bmatrix} 0 & -gx & -gy & -gz \\ gx & 0 & gz & -gy \\ gy & -gz & 0 & gx \\ gz & gy & -gx & 0 \end{bmatrix} Qtn$

$recipNorm \leftarrow 1 / \sqrt{q0^2 + q1^2 + q2^2 + q3^2}$

$Qtn \leftarrow recipNorm \cdot Qtn$

$Qtn_filter = Qtn$

- Timer1 FastPWM, Timer3 CTC
- ADC(Analog to Digital Converter) 통신으로 Atmega128 <- GP2Y0A02(적외선 거리센서)
- 각도 -> OCR 변환

$$OCR = OCR_{\min} + \frac{\text{deg}}{180.0^\circ} (OCR_{\max} - OCR_{\min})$$

5. 중간 보고

팀 활동 보고서

미팅 개요	일시	2025.11.18.(화)		
	참석자	최지현		
	본 미팅의 주요활동	Servo 모터를 동작하기 위해서 ATmega128의 PWM 생성		
진행 사항	구분	활동 내용		조치 사항
	이번 미팅에서 한 일	ATmega128에서 Fast PWM으로 20ms의 주기를 생성하기 위한 prescaler 설정과 Servo 모터 2개를 제어하기 위해 Timer1의 OCR1A, OCR1B핀으로 연결함.		서보모터 각도를 확인하기 위해 duty cycle 0.5ms~2.0ms를 확인함
	논의 사항	Datasheet 상으로 duty cycle 1ms가 0도, 2ms가 180도인데 MG996R 서보모터는 0.5ms가 0도, 2.5ms가 180도 임을 확인함		
추후 계획	구분	활동 내용	역할 분담	추진 일정
	다음 미팅에서 해야 할 일	MPU6050 센서값 확인 및 calibration 알고리즘 작성		25.11.25.(화)
	기타			
피드백	교수님의 피드백 사항	<p>서버 모터의 동작은 데이터 시트에 나온 대로 동작 확인한 것으로 실제 구현상의 문제점이나 개선 사항을 논의하기 바람.</p> <p>본 프로젝트의 제목을 명명하여 계획서에 반영하기 바람.</p> <p style="text-align: right;">수업 담당 교수: 이 용 </p>		

팀 활동 보고서

미팅 개요	일시	2025.11.25.(화)		
	참석자	최지헌		
	본 미팅의 주요활동	GY-25의 구동방식과 Pin 설정, 초기 설정 확인		
진행 사항	구분	활동 내용		조치 사항
	이번 미팅에서 한 일	GY-25를 구동하기 위해 데이터시트를 참조하여 GY-25의 레지스터를 확인하고 부품 연결을 위해 납땜을 함. TWI 통신으로 하려고 했지만 통신이 되지 않았음		TWI통신이 되지 않아 UART 통신으로 변경
	논의 사항	TWI로 직접 통신할 예정이었으나 UART 통신으로 바꾸면서 다른 센서(적외선 센서)를 어디와 연결할지 생각해 보아야 함		
추후 계획	구분	활동 내용	역할 분담	추진 일정
	다음 미팅에서 해야 할 일	MPU6050 센서 연결, 값 확인, 초기화 실행 및 인터페이스 작성		25.12.02(화)
	기타			
피드백	교수님의 피드백 사항	수업 담당 교수: 이 응 혁 (인)		

팀 활동 보고서

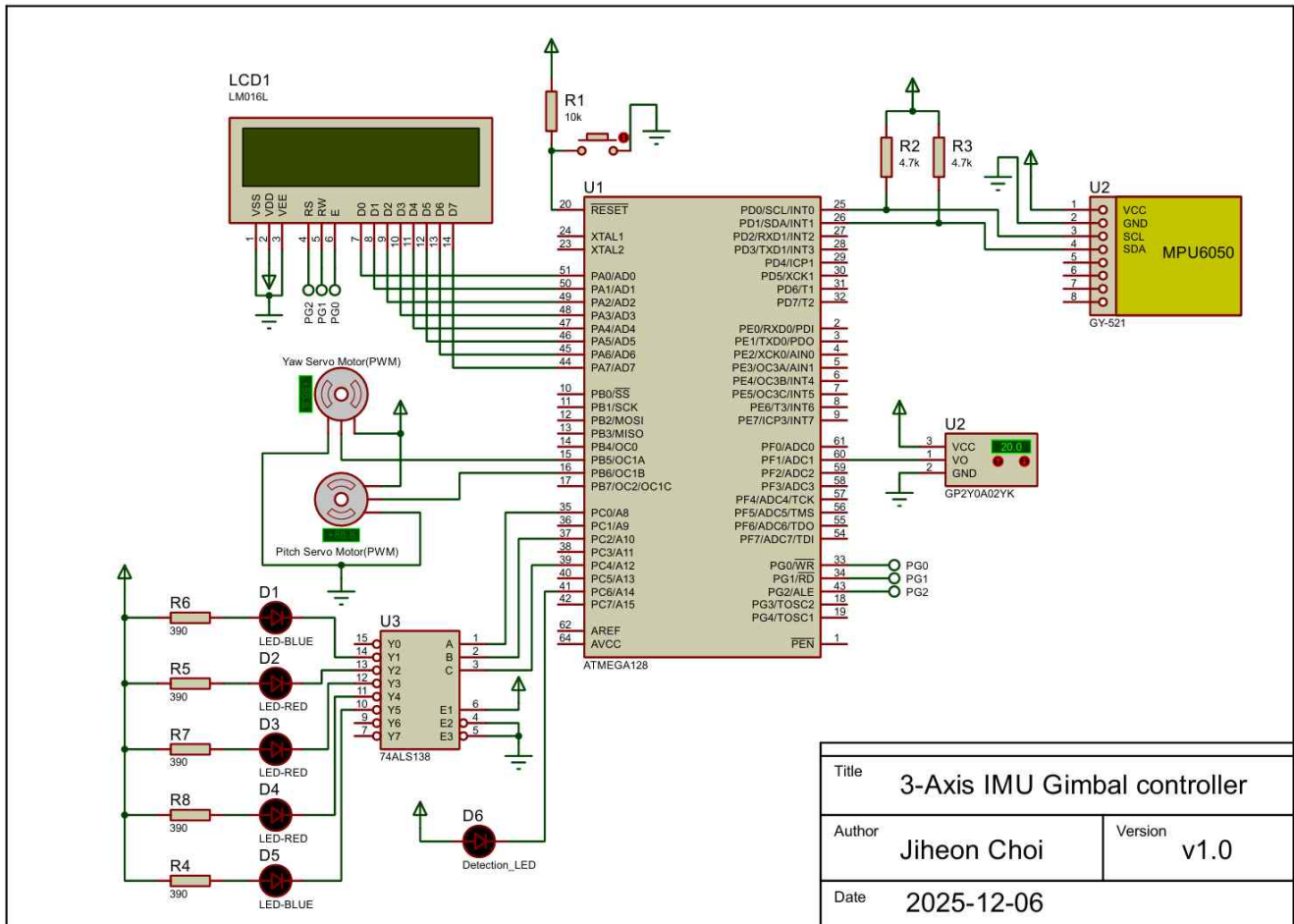
미팅 개요	일시	2025.12.02.(화)		
	참석자	최지헌		
	본 미팅의 주요활동	GY-25(MPU6050)의 각도 값을 Servo모터의 제어에 사용		
진행 사항	구분	활동 내용		조치 사항
	이번 미팅에서 한 일	GY-25로 각도를 읽어 LCD에 출력함. GY-25의 Yaw, Pitch로 2개의 Servo모터의 움직임을 제어하려고 했으나 너무 작은 움직임까지 각도에 반영되어 진동하듯이 움직였고 GY-25의 각도값이 정확하지 않았음		GY-25의 각도를 읽을 때 Low Pass Filter를 사용해서 작은 각도 변화를 무시함
	논의 사항	작은 각도 변화를 무시했으나 각도값이 튀는 상황이 발생하여 Servo가 급격하게 변화함. 부드럽게 제어할 수 있도록 이상치를 제거하고 실제 각도랑 맞도록 보정해야 함		
추후 계획	구분	활동 내용	역할 분담	추진 일정
	다음 미팅에서 해야 할 일	GY-25의 이상치를 제거하고 실제 각도랑 같은 값이 되도록 보정		25.12.09.(화)
	기타			
피드백	교수님의 피드백 사항	수업 담당 교수: 이 응 혁 (인)		

팀 활동 보고서

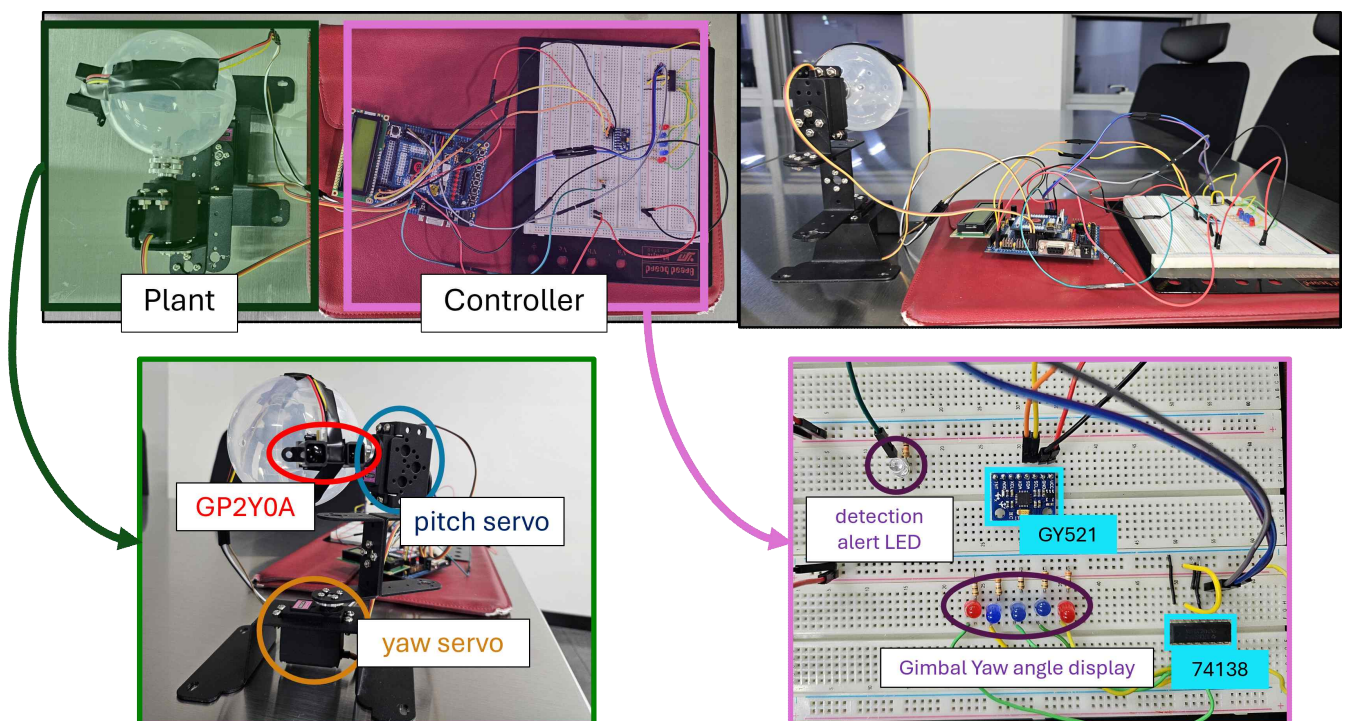
미팅 개요	일시	2025.12.09.(화)		
	참석자	최지현		
	본 미팅의 주요활동	GY-521(3축 IMU)과 TWI 통신으로 3축 각도를 읽고, 읽은 각도로 Servo motor 제어, GP2Y0A02YK0F(적외선 거리센서)로 객체 탐지		
진행 사항	구분	활동 내용		조치 사항
	이번 미팅에서 한 일	GY-521(3축 IMU)을 TWI 통신으로 3축 각도를 읽고 절대 각도 값으로 servo 모터를 제어하려 했으나 drift로 인해 정밀 제어가 불가능 하였음		상대 각변위 알고리즘을 개발하여 3축 IMU의 각 변화량을 서보모터의 제어에 사용함
	논의 사항	3축 IMU가 가진 고질적인 drift문제를 현 시점의 각 변위량으로 servo모터를 제어하여 지자기 센서 없이 IMU의 drift문제를 해결하였음		
추후 계획	구분	활동 내용	역할 분담	추진 일정
	다음 미팅에서 해야 할 일	발표		25.12.19
	기타			
피드백	교수님의 피드백 사항	<p style="text-align: right;">수업 담당 교수: 이 응 혁 (인)</p>		

6. 수행 내용

회로도



하드웨어 구성 사진



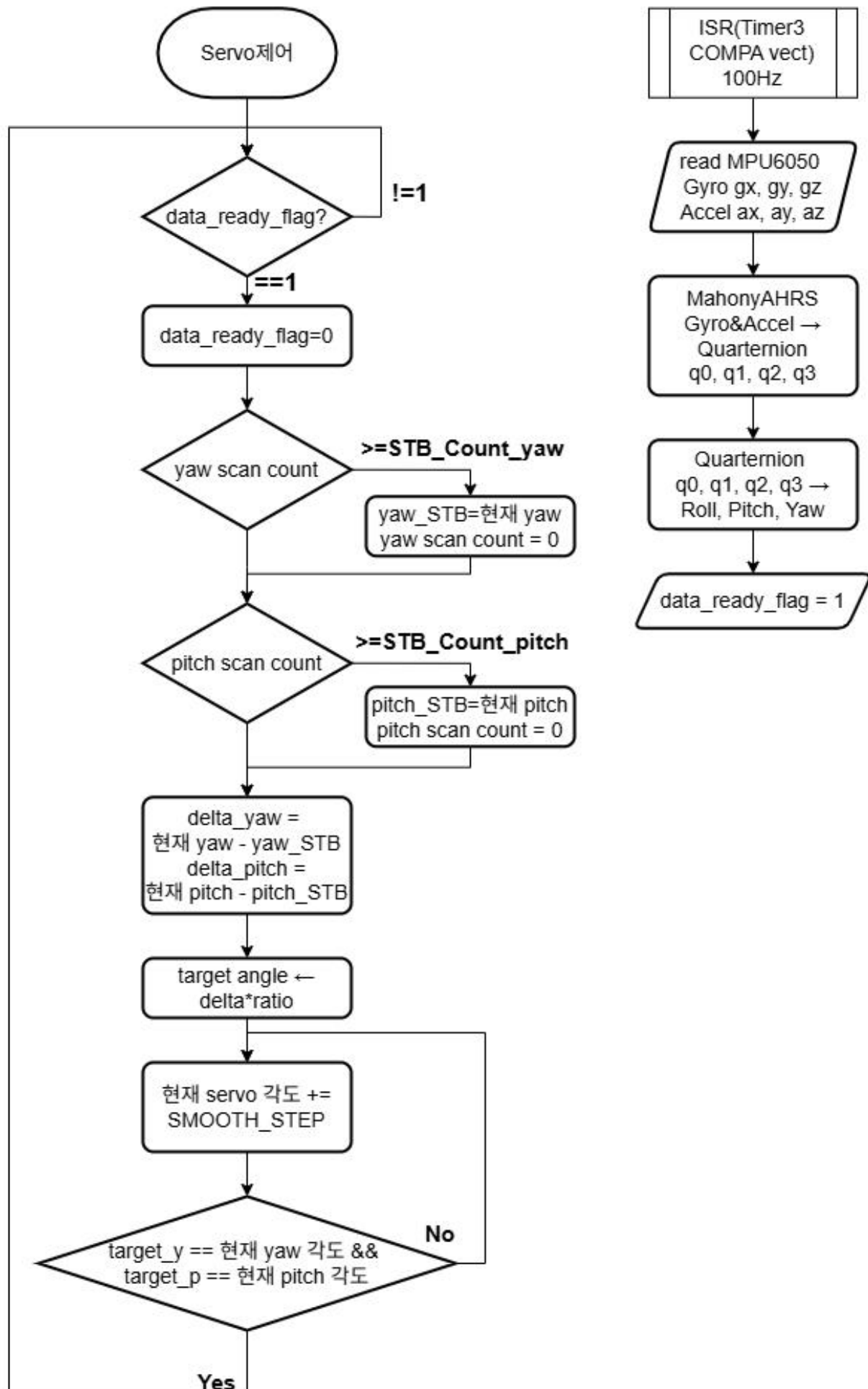
수행 목록표

N	개발목표	해결 방법	완성도
1	3축 IMU 센서에서 변하는 Yawing, Pitching 동작들을 Atmega128로 읽는다.	I2C(TWI) 통신을 사용했으며 Timer/Counter 1의 CTC모드로 설정해서 100Hz로 Sampling 되도록 함. 오일러각도로 변환하는 과정에서 drift를 줄이기 위해서 MPU6050에서 LPF, HPF를 설정하고, Mahony 알고리즘으로 쿼터니언에서 오일러 각도로 변환했다.	100%
2	MPU6050의 누적 오차를 잡기 위해 MPU6050을 주기적으로 재시작&Calibration 한다	Sampling 횟수가 각각 pitch는 2번, yaw는 5번 이상일 때 그때의 pitch와 yaw 각도를 기준 각도로 두고 다음 Sampling 되는 각도와와의 차이 $\Delta\theta$ 을 서보모터의 제어에 사용하였으므로 재시작&Calibration을 할 필요가 없었다.	100%
3	서보모터가 MPU6050의 10°미만의 작은 움직임을 제대로 추종할 수 없으므로 단위 각도 이상인 움직임만 추종한다.	서보모터 제어를 위한 $\Delta\theta$ 를 입력하였을 때 비례이득 K값을 곱해서 작은 움직임도 추종할 수 있도록 했으며, $\Delta\theta=1$ 은 drift 각도일 위험이 크기 때문에 $\Delta\theta>1$ 이상일 때만 제어 되도록 했다.	100%
4	Atmega128에서 읽은 가속도 센서의 변화량을 일정 비율의 단위 각도로 변환하여 서보모터 2개로 구(Sphere)형 짐벌(Gimbal)을 움직여서 짐벌이 3축 IMU를 추종하도록 한다	서보모터의 현재 각도를 알고 있는 상태에서 목표각도 = 현재각도 + (MPU6050입력 K* $\Delta\theta$)를 계산하여 목표각도를 OCR 값으로 변환해서 PWM의 duty cycle을 변경하여 모터의 각도를 제어함. Yaw서보모터의 각도 범위를 5단계로 분할 하여 5개디 LED로 서보모터가 Yaw 방향으로 회전했는지 표시한다.	100%
5	짐벌에 부착된 적외선 거리 센서로 짐벌이 바라보는 방향에 있는 물체를 감지한다.	적외선 거리 센서를 구형 짐벌 정면에 부착하고 ADC로 적외선 거리센서의 출력 전압을 데이터 시트에 있는 출력전압-거리 관계의 표를 참고하여 거리를 계산하였고 물체가 50cm 이내에 위치한다면 3색 LED가 점등 되도록 함	100%

수행 과정 중 문제점 및 해결 사항

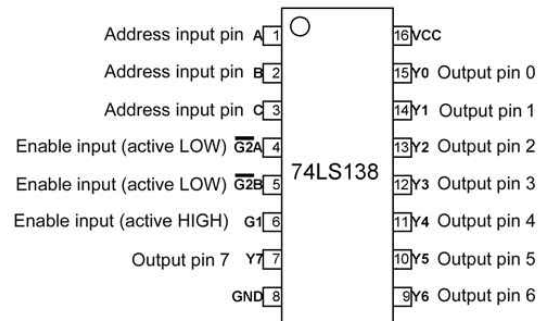
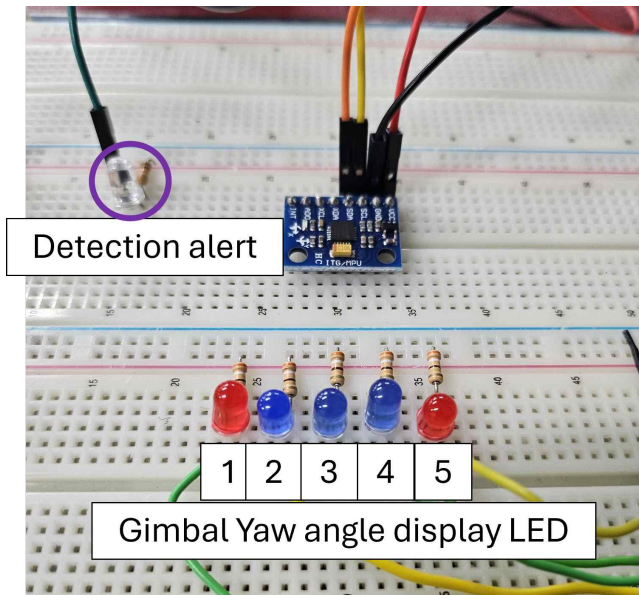
- 실습 보드에 대한 Pin map이 구체적으로 되어 있지 않아서 정확한 Pin을 찾는데 시간을 사용했으며 강의자료에는 USART1 단자가 실제로 USART0인 경우가 있었음
- Servo Motor가 data sheet와 다른 duty cycle로 움직이는 것이 문제였으나 duty cycle를 스동으로 조절하여 각도를 찾음
- GY-25를 UART로 사용하다가 고장나는 바람에 GY-521을 TWI통신으로 사용하게 됨. register map과 제품 사양서가 따로 있어서 I2C 통신을 설정하기 위해 아두이노 코드도 함께 참조하여 register map과 제품 사양서를 통합해서 보는 방법을 학습함

서보모터 제어 알고리즘 흐름도



LED 인터페이스 제어

- DeMUX 74138 사용



범위	C (PC4)	B (PC2)	A (PC0)	점등 LED
-60°~-40°	L	L	H	1
-39°~-20°	L	H	L	2
-19°~20°	L	H	H	3
21°~-40°	H	L	L	4
41°~60°	H	L	H	5

- PC4 – 74138의 입력 C
- PC2 – 74138의 입력 B
- PC0 – 74138의 입력 A
- gimbal의 현재 Yaw 각도를 사용자에게 편하게 알려주기 위해서 사용
- Detection alert LED는 적외선 거리센서의 거리 $L[\text{cm}] < 50\text{cm}$ 이내 일 때 점등

7. 계획 대비 달성도 평가

- 수행 목록표에 나온대로 모든 목표를 달성했음
- 서보모터를 연속 시간으로 제어하려고 하다 보니 갑자기 변화하는 각도에 대해서 대응 하기 위해 서보모터가 계단식으로 움직임, 멀티 프로세스나 멀티 스레드를 추가했다면 IMU 를 읽는 스레드 따로 Servo 움직이는 스레드 따로 나눠서 목표 각도로 움직이는 중에도 다른 목표 각도로 움직이도록 변경할 수 있음

8. 학습자 성찰

- 맨 마지막 장 <학습자 성찰 일지>로 대체함.

9. 맺음말

- 하나의 시스템 설계를 하드웨어 설정부터 하는 임베디드 개발의 프로세스를 간단하게 배울 수 있어서 뜻깊은 수업이었다. 전자공학부라면 응당 들어야 하는 수업이다.
- FreeRTOS를 사용하는 교과가 있었으면 한다.

학습자 성찰 일지

학번	2021146036	이름	최지현
1. 이번 학습을 통해 무엇을 배웠나요?			
임베디드 프로세서를 이용해 제품(시스템)을 만들기 위해 하드웨어를 결정하고 데이터 시트를 분석하고 안정적인 제어를 위해 알고리즘을 개발하고 하드웨어&소프트웨어 디버깅하는 방법을 배웠다.			
2. 수업에서 어려웠던 활동은 무엇이었나요?			
부품을 새로 구매했을 때 데이터 시트를 분석해서 레지스터를 설정하는 것이 어려웠습니다.			
3. 앞으로 내가 더 알고 싶은 내용은 무엇인가요?			
프로세서에 pthread, FreeRTOS를 마운트해서 멀티 프로세스가 가능하도록 하는 프로젝트를 할 것이며 피지컬 AI를 접목 시킬 예정이다.			
4. 학습한 내용을 적용할 수 있는 것은 무엇인가요?			
하드웨어가 포함된 전자기기들을 제어하는 모든 분야에서 사용 할 수 있다.			
5. 이 수업에서 나의 부족한 부분은 무엇인가요?			
다른 프로젝트와 병렬로 하는 바람에 시간이 부족했다.			
6. 이 수업의 학습과정을 통해 무엇을 느꼈나요?			
임베디드 개발은 재밌고 낭만 넘치는 직업이다. 내가 원하는 시스템을 내 손으로 만들 수 있다는 즐거움을 얻을 수 있다.			
7. 기타 느낀 점을 자유롭게 기술하세요.			
교수님께서 고안해주신 커리큘럼으로 한층 더 성장할 수 있었습니다. 감사합니다 교수님. 더 열심히 배워서 산업계에서 인정받는 엔지니어가 되도록 하겠습니다.			