

임베디드 응용 및 실습 과제

8주차. 과제 보고서

과 목 명	임베디드응용및실습
학 번	2021161104
이 름	이주호
제 출 일	2025년 10월 31일

1-1 실습문제

- lecture 7 lab 강의 자료 맨 마지막 페이지의를 수행
- 작성 코드와 실행 장면을 동영상으로 짧게 촬영
- 자신의 github의 폴더를 생성하여 보고서(소스 코드)와 동영상을 업로드하고 URL만란에 제출

1-2. 소스코드

```
# -*- coding: utf-8 -*-
import RPi.GPIO as GPIO
import serial
import threading
import time

# =====
# 핀 설정
# =====

PWMA =18
PWMB =23
AIN1 =22
AIN2 =27
BIN1 =25
BIN2 =24

# =====
# GPIO 초기화
# =====

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(PWMA, GPIO.OUT)
GPIO.setup(PWMB, GPIO.OUT)
GPIO.setup(AIN1, GPIO.OUT)
GPIO.setup(AIN2, GPIO.OUT)
GPIO.setup(BIN1, GPIO.OUT)
GPIO.setup(BIN2, GPIO.OUT)

# =====
# PWM 설정 (주파수 250Hz)
# =====

L_Motor = GPIO.PWM(PWMA, 250)
R_Motor = GPIO.PWM(PWMB, 250)
L_Motor.start(0)
R_Motor.start(0)

# =====
# 블루투스 시리얼 설정
# =====

bleSerial = serial.Serial("/dev/ttyS0", baudrate=9600, timeout=1.0)
```

```

# =====
# 전역 변수
# =====
gData = ""
prev_cmd = None
# =====
# 모터 제어 함수 (버튼 기반 구조 유지)
# =====
def stop_motor():
    GPIO.output(AIN1, 0)
    GPIO.output(AIN2, 0)
    GPIO.output(BIN1, 0)
    GPIO.output(BIN2, 0)
    L_Motor.ChangeDutyCycle(0)
    R_Motor.ChangeDutyCycle(0)
    print("▶ 정지")

def go_forward():
    print("▶ 전진")
    GPIO.output(AIN1, 0)
    GPIO.output(AIN2, 1)
    GPIO.output(BIN1, 0)
    GPIO.output(BIN2, 1)
    L_Motor.ChangeDutyCycle(50)
    R_Motor.ChangeDutyCycle(50)

def turn_left():
    print("▶ 좌회전")
    GPIO.output(AIN1, 0)
    GPIO.output(AIN2, 1)
    GPIO.output(BIN1, 0)
    GPIO.output(BIN2, 1)
    L_Motor.ChangeDutyCycle(25)
    R_Motor.ChangeDutyCycle(50)

def turn_right():
    print("▶ 우회전")
    GPIO.output(AIN1, 0)
    GPIO.output(AIN2, 1)
    GPIO.output(BIN1, 0)
    GPIO.output(BIN2, 1)
    L_Motor.ChangeDutyCycle(50)
    R_Motor.ChangeDutyCycle(25)

def go_backward():
    print("▶ 후진")

```

```

GPIO.output(AIN1, 1)
GPIO.output(AIN2, 0)
GPIO.output(BIN1, 1)
GPIO.output(BIN2, 0)
L_Motor.ChangeDutyCycle(50)
R_Motor.ChangeDutyCycle(50)
# =====
# 블루투스 수신 스레드 (에러 방지 처리 포함)
# =====
def serial_thread():
    global gData
    while True:
        if bleSerial.in_waiting > 0:
            try:
                # 잘못된 바이트(0xA5 등)로 인한 decode 에러 방지
                data = bleSerial.readline().decode('utf-8', errors='ignore').strip()
                if data != "":
                    gData = data
                    print("수신 데이터:", gData)
            except Exception as e:
                print("디코딩 오류:", e)
# =====
# 메인 루프 (조이스틱 각도 기반 제어)
# =====
def main():
    global gData, prev_cmd
    try:
        while True:
            if gData.startswith("J0:"):
                try:
                    # 예: "J0:180,0.6"
                    data = gData.replace("J0:", "")
                    angle_str, power_str = data.split(",")
                    angle = float(angle_str)
                    power = float(power_str)
                    gData = ""
                    # 강도(power)가 너무 낮으면 정지
                    if power < 0.2:
                        if prev_cmd != "stop":
                            stop_motor()
                            prev_cmd = "stop"
                        continue
                except:
                    pass

```

```

# 각도 범위별 동작
if 45 <= angle <135:
    if prev_cmd != "go":
        go_forward()
        prev_cmd = "go"
elif 135 <= angle <225:
    if prev_cmd != "left":
        turn_left()
        prev_cmd = "left"
elif 225 <= angle <315:
    if prev_cmd != "back":
        go_backward()
        prev_cmd = "back"
else: # 나머지 (315~360, 0~45)
    if prev_cmd != "right":
        turn_right()
        prev_cmd = "right"
except ValueError:
    # 데이터 파싱 실패 시 무시
    print("데이터 파싱 오류:", gData)
    gData = ""
except Exception as e:
    print("예외 발생:", e)
    gData = ""
    time.sleep(0.05)
except KeyboardInterrupt:
    print("\n사용자 중단 - 프로그램 종료")
    stop_motor()
    GPIO.cleanup()

# =====
# 프로그램 시작
# =====
if __name__ == '__main__':
    print("Bluetooth 조이스틱 각도 기반 자동차 제어 시작")
    stop_motor()
    t1 = threading.Thread(target=serial_thread)
    t1.daemon = True
    t1.start()
    main()
    bleSerial.close()
    GPIO.cleanup()
    print("프로그램 종료")

```

1-3. 실행결과

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL
수신 데이터: J0:238,0.0
수신 데이터: J0:1,0.6
▶ 우회전
수신 데이터: J0:1,0.8
수신 데이터: J0:1,0.8
수신 데이터: J0:0,0.0
▶ 정지
수신 데이터: J0:219,0.0
수신 데이터: J0:179,0.9
▶ 좌회전
수신 데이터: J0:0,0.0
▶ 정지
수신 데이터: J0:136,0.1
수신 데이터: J0:269,0.5
▶ 후진
수신 데이터: J0:269,0.8
수신 데이터: J0:0,0.0
▶ 정지
□
```