

2024 임베디드프로세서2

Term-Project 보고서

소 속 전자공학부 / 임베디드시스템 전공

학 번 2019146037

성 명 홍석영

제출일 2024.12.21

목 차

1 프로젝트 필요성	02-03
2 프로젝트 개요	04-06
3 수행계획서	07
4 주요 핵심 기술	08-10
5 중간 보고	11
6 수행 내용	12-15
7 계획 대비 달성도 평가	16
8 학습자 성찰	17
9 맺은말	18

1. 프로젝트 필요성

프로젝트명: 택배 도난 방지 보관함

개발 배경

2020년 코로나19 팬데믹 이후, 우리의 일상생활과 사회적 환경은 급격한 변화를 맞이했습니다. 특히 택배 배송 방식에서도 큰 변화가 나타났습니다. 기존의 대면 배송은 택배 기사와 수령인이 직접 만나 물품을 전달받는 방식이었으나, 감염 위험을 최소화하기 위해 비대면 배송이 빠르게 확산되며 새로운 표준으로 자리 잡았습니다.

비대면 배송은 택배 기사와 수령인 간의 시간적·공간적 제약을 해소해주는 장점이 있습니다. 수령인이 집에 없더라도 택배 기사가 물품을 문 앞이나 지정된 장소에 놓고 가는 방식으로, 효율성과 편리성을 제공해 바쁜 현대인의 일상에 적합한 서비스로 평가받고 있습니다.

그러나 이러한 편리함의 이면에는 심각한 문제점이 존재합니다. 비대면 배송은 물품이 외부에 장시간 노출되는 구조적 한계를 가지며, 이는 택배 물품 도난이라는 사회적 문제로 이어지고 있습니다.

- 코로나19 팬데믹 이후 택배 물품 도난 사례는 급증했으며, 이는 택배 수령자들의 불안감을 증폭시키고 있습니다.

- 또한, 택배 물품이 도난당했을 경우 책임 소재가 불분명하여 소비자와 택배 회사 간 갈등이 증가하는 등 민원과 피해 사례가 지속적으로 보고되고 있습니다.

이뿐만 아니라, 택배를 매개로 한 신종 범죄 수법도 등장하고 있습니다.

- 범죄자들은 택배 기사로 위장하여 수령인의 집 앞에서 택배를 훔쳐가는 것은 물론, 이를 기회로 가정에 침입해 더 큰 범죄를 저지르는 사례도 발생하고 있습니다.

- 이러한 택배 기사 변장 범죄는 단순한 물품 도난을 넘어 수령인의 개인 안전과 가정의 보안을 심각하게 위협하는 사회적 문제로 대두되고 있습니다.

이러한 문제들은 비대면 배송의 편리함을 지속적으로 누리기 위해 반드시 해결해야 할 과제로 인식되고 있습니다. 택배를 안전하게 보관하고, 도난 및 범죄의 위험을 효율적으로 차단할 수 있는 시스템의 필요성이 점점 더 강조되고 있습니다. 이에 본 프로젝트에서는 '택배 도난 방지 보관함'을 개발하고자 합니다.

프로젝트 목표

본 프로젝트의 최종 목표는 비대면 배송의 편리함을 유지하면서도 택배 도난 및 범죄로부터 안전을 보장하는 시스템을 구현하는 것입니다. 이를 위해 개발되는 택배 도난 방지 보관함은 다음과 같은 주요 목표를 달성하고자 합니다:

1. 안전한 택배 보관

- 보관함 내부에 택배 물품을 보관하여 외부 노출을 최소화합니다.
- 물품이 보관된 상태에서 무단으로 접근하거나 보관함을 열려고 시도할 경우, 경고음을 발생시켜 도난 가능성을 줄이고 즉각적인 대처가 가능하도록 설계합니다.

2. 보안 강화

- 비밀번호 입력 방식을 적용하여 보관함의 무단 접근을 방지합니다.
- 3회 이상 비밀번호 입력이 실패할 경우 경고음을 울리고 원격지 사용자에게 경고 알림을 전송하여 보안을 강화합니다.
- 택배를 통해 발생할 수 있는 범죄를 예방하고, 사용자와 가정의 안전을 보장합니다.

3. 사용자 편의성 제공

- Keypad, LCD, FND와 같은 직관적인 사용자 인터페이스를 제공하여 사용자가 쉽게 시스템을 조작하고 현재 상태를 확인할 수 있도록 설계합니다.
- 보관함 상태와 알림을 시각적(LCD, FND) 및 청각적(Buzzer)으로 동시에 제공하여 사용성을 극대화합니다.
- 관리자는 버튼을 통해 관리자 모드에 진입하여 비밀번호를 변경할 수 있어 보관함을 지속적으로 안전하게 유지할 수 있습니다.

4. 스마트 기능 구현

- 초음파 센서를 통해 물품 보관 상태를 실시간으로 감지하여 사용자가 택배 물품이 보관되었는지 여부를 명확히 확인할 수 있도록 합니다.
- 비밀번호 인증이 완료되면 서보 모터를 활성화하여 보관함을 열고 닫는 동작을 자동으로 수행함으로써 자동화된 보안 관리를 제공합니다.

본 프로젝트는 단순히 보관함의 역할을 넘어, 택배를 매개로 발생할 수 있는 다양한 위험 요소를 효과적으로 방지하고 사용자가 안심할 수 있는 스마트 보안 솔루션을 제공하는 데 초점을 맞추고 있습니다.

2. 프로젝트 개요

2-1) 최종 개발 목표

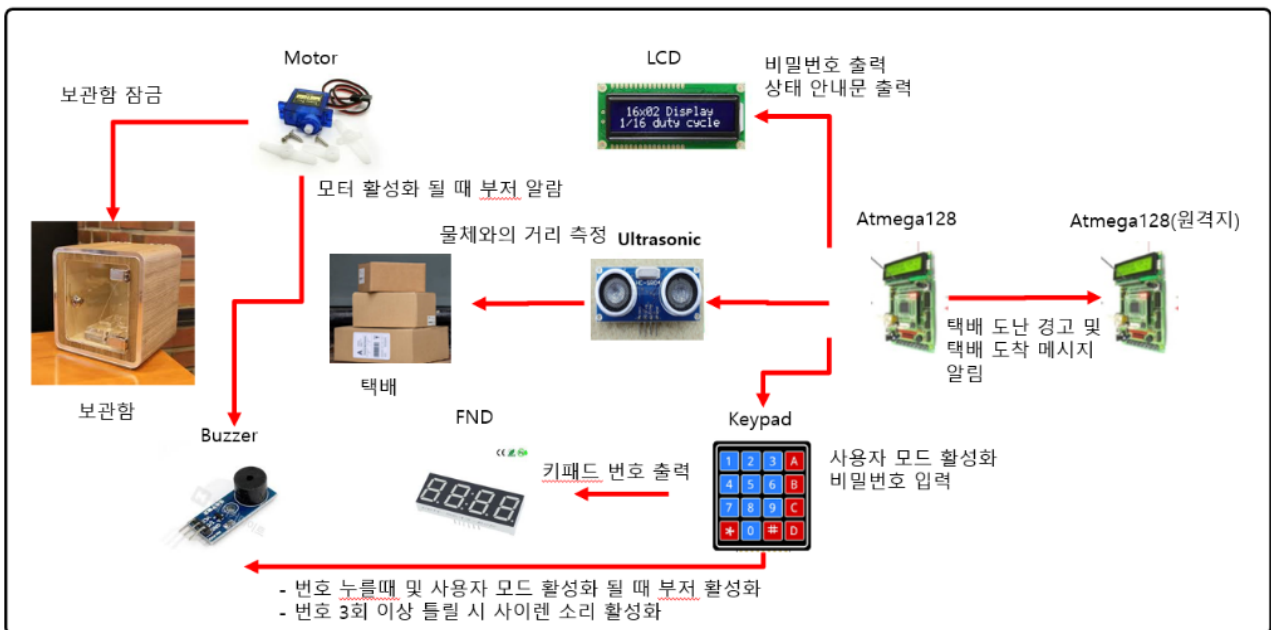
최종 개발 목표:

- 도난 및 범죄를 예방하는 무인 택배 보관함

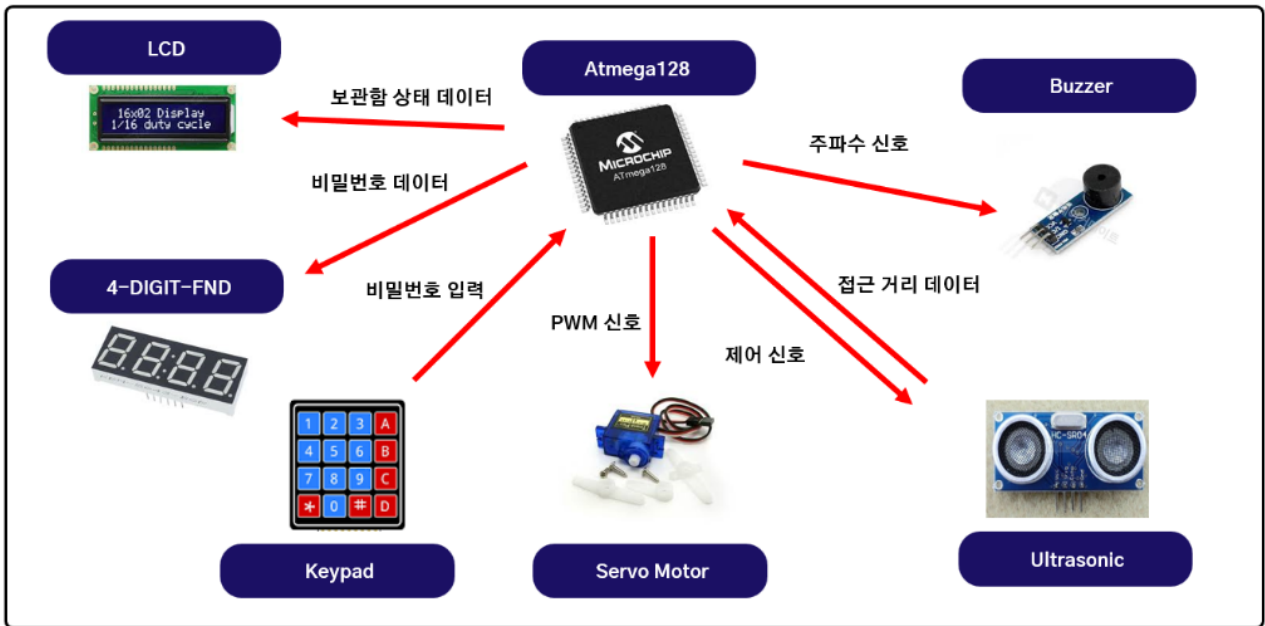
세부 개발 목표:

- 택배 보관
- 택배 도난 알람
- 관리자 모드를 통한 비밀번호 변경
- 사용자 친화적 인터페이스 제공 (LCD, FND)
- 청각적 정보 제공 (Buzzer)

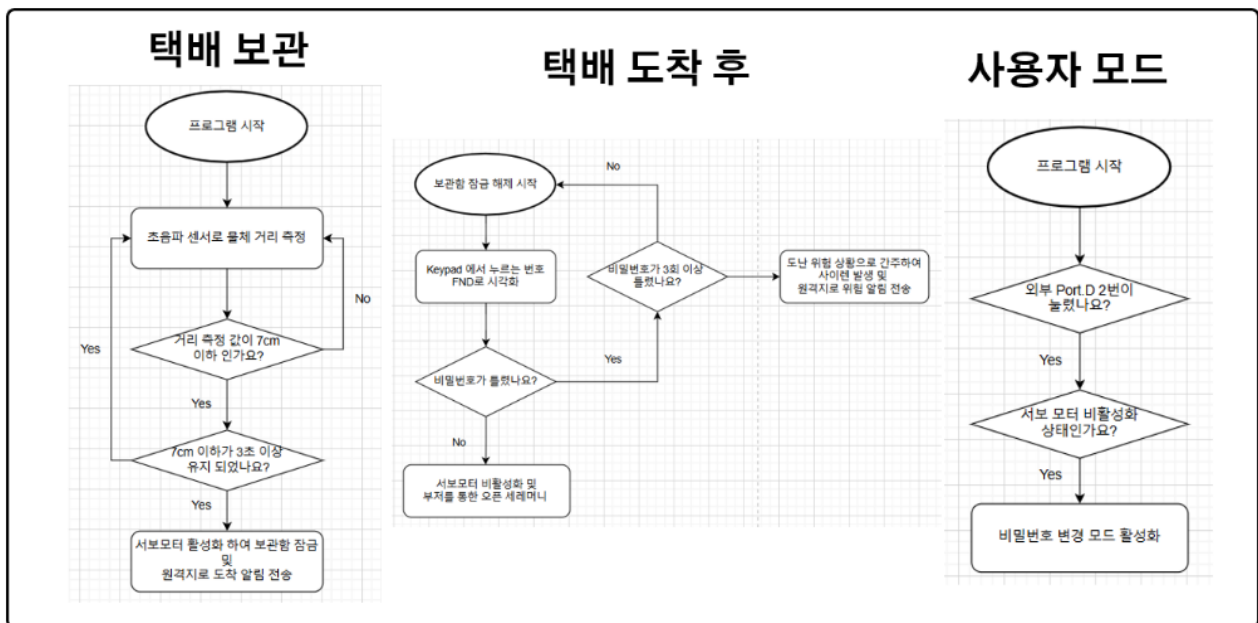
2-2) 시스템 개념도



2-3) HW 구성도



2-4) SW 시퀀스



1. 택배 보관

- 서보 모터: 초기 활성화 상태
- 비밀번호 입력: Keypad를 통해 입력
- 인증 과정: 비밀번호 일치 시 서보 모터 비활성화, 초음파 센서 활성화
- 택배 감지: 초음파 센서가 물체와의 거리가 7cm 이하일 경우, 택배가 보관된 것으로 인식

- 잠금 상태: 서보 모터 다시 활성화하여 보관함 잠금
- 도착 알람 전송: 원격지 사용자에게 택배 도착 알림 전송

2. 도난 상황

- 비밀번호 오류: 서보 모터 활성화 상태에서 비밀번호 3회 이상 오입력 시 경고음 발생
- 경고음 지속: 올바른 비밀번호 입력 전까지 경고음 지속
- 경고 메시지 전송: 원격지 사용자에게 경고 메시지 전송

3. 사용자 모드

- 모드 실행 조건: 서보 모터 비활성화 상태에서 PORTD.2 버튼이 눌리면 사용자 모드 실행
- 보안 강화: FND 비활성화 후 Keypad를 통해 새 비밀번호 입력
- 비밀번호 변경: '#' 버튼 입력 시 새로운 비밀번호로 변경 완료
- 서보 모터 활성화: 비밀번호 변경 후 서보 모터를 다시 활성화

3. 팀별 과제수행계획서 (문제 분석지)

모듈명	스마트 택배 도난 방지 보관함		
팀 이름	스마트 택배 도난 방지 보관함	날짜	2024.11.21
팀원	홍석영		

학습 목표	임베디드 시스템 개발을 위한 제품 필요성, 시스템 구성(H/W 및 S/W), 시스템 개발을 위한 전체적인 제반사항의 목표 설정을 바탕으로 마이크로컨트롤러를 활용한 GPIO, Interrupt, Timer/Counter, 직렬통신(uart) 융합을 통한 임베디드 시스템 개발 역량 강화
----------	--

가설/해결안 (Ideas)	알고 있는 사실 (Facts)	더 알아야 할 사항 (Learning Issues)	앞으로의 계획 (Action Plan)
<ul style="list-style-type: none"> - 택배 도난율 및 택배기사 변장 범죄율이 점차 증가함에 따라 택배 도난 방지 보관함을 설치하여 범죄를 방지 	<ul style="list-style-type: none"> - 택배 보관함을 통해 택배 도난 및 택배기사 변장 범죄율 감소 - ATmega128 보드의 레지스터 및 GPIO 핀 구성 요소 	<ul style="list-style-type: none"> - 초음파 센서(HC-SR04) 제어 방법 - FastPWM을 통한 ServoMotor 제어 방법 	<ul style="list-style-type: none"> - 주제 선정 구체화 진행 - HW 설계 및 센서 관련 GPIO 핀 설정 - SW 순서도를 구성후 알고리즘 개발 - 센서 제어 방법 탐구

교수 comment	
---------------	--

4. 주요 핵심 기술

4-1) LCD

LCD는 보관함의 상태를 사용자에게 시각적으로 전달하기 위해 사용됩니다.

- 주요 기능으로는 Keypad를 통해 입력된 비밀번호 표시, 초음파 센서를 통해 측정된 거리 정보, 그리고 시스템의 전반적인 동작 상태를 시각적으로 표시하는 역할을 합니다.
- 특히, LCD는 사용자가 비밀번호 입력 과정을 확인하거나 현재 시스템이 잠금 상태인지, 관리 모드인지 등을 손쉽게 파악할 수 있도록 도와줍니다.
- 이러한 사용자 친화적 인터페이스는 시스템의 직관성과 사용 편의성을 높이는데 기여하며, 프로젝트의 핵심 디스플레이 장치로 선정되었습니다.

4-2) Keypad

Keypad는 비밀번호 입력 및 관리 모드에서 주요 입력 장치로 사용됩니다.

- 4x4 Keypad를 통해 0~9의 숫자 키와 특수 키('*', '#')를 지원하며, 다양한 비밀번호 조합을 생성할 수 있도록 설계되었습니다.
- 비밀번호 인증 과정에서 사용자는 Keypad를 이용해 입력하며, 입력된 비밀번호는 LCD 및 FND를 통해 실시간으로 확인할 수 있습니다.
- Keypad는 간단한 구조와 높은 신뢰성을 바탕으로 시스템의 입력 인터페이스로 적합하며, 사용자가 직관적으로 조작할 수 있는 장점이 있습니다.

4-3) FND (7-Segment Display)

FND는 비밀번호 입력 시 실시간으로 숫자를 시각화하여 사용자에게 입력 과정을 더 명확하게 보여줍니다.

- LCD의 작은 화면으로 인해 시력이 약한 사용자, 특히 어르신들이 비밀번호를 입력할 때 어려움을 겪을 수 있다는 점을 고려하여 추가되었습니다.
- 4자리 FND를 사용함으로써 Keypad로 입력한 숫자가 크고 명확하게 표시되어 사용자 편의성이 향상됩니다.
- LCD와 함께 작동하여 입력 중 오류를 줄이고, 시스템의 시각적 정보 제공 능력을 보완합니다.

4-4) Buzzer

Buzzer는 시스템의 다양한 상태를 청각적으로 알리는 기능을 제공합니다.

주요 용도는 다음과 같습니다

- Keypad 입력 확인음: 사용자가 Keypad를 눌렀을 때 입력이 정상적으로 인식되었음을 알림.
- 비밀번호 오류 경고음: 비밀번호 입력 오류 시 경고음을 발생시켜 잘못된 입력을 즉시 인지할 수 있도록 지원.
- 보관함 잠금/해제 알림음: 잠금 상태 전환 시 사용자에게 알림.
- 시각 정보가 부족하거나 즉각적인 반응이 필요한 경우, Buzzer를 통해 효율적인 알림 시스템을 구현할 수 있습니다.
- 이를 통해 시스템은 시각적, 청각적 정보를 동시에 제공하며, 사용자가 더 직관적으로 시스템을 이해할 수 있도록 설계되었습니다.

4-5) HC-SR04 초음파 센서

초음파 센서 HC-SR04는 보관함 내부에 택배 물품이 있는지 여부를 판단하는 역할을 합니다.

- HC-SR04는 2cm에서 4m까지의 탐지 거리를 지원하며, 높은 측정 정확도와 광범위한 거리 탐지 능력을 갖추고 있습니다.
- 보관함 내부에 물체가 들어왔을 때, 초음파 센서는 물체와의 거리를 측정하고 이를 통해 물품 보관 여부를 시스템에 전달합니다.
- 초음파 센서는 Keypad 입력 후 물품이 정상적으로 보관된 경우에만 서보 모터를 다시 활성화하여 잠금을 수행하도록 설계되어 있습니다.
- 이 기술은 사용자가 안전하고 효율적으로 택배를 보관할 수 있도록 지원하며, 도난 방지의 핵심적인 역할을 수행합니다.

4-6) Servo Motor

서보 모터는 보관함의 잠금/해제 기능을 수행하는 핵심 부품입니다.

- Keypad를 통해 올바른 비밀번호가 입력되면 서보 모터가 활성화되어 잠금 장치가 해제됩니다.
- 보관함에 물품이 보관된 후, 다시 서보 모터가 활성화되어 보관함을 잠그는 작업을 수행합니다.
- 서보 모터는 정확하고 안정적인 작동이 가능하며, 프로젝트의 보안성을 강화하는 필수 부품입니다. 또한, 작은 크기와 낮은 전력 소비량으로 프로젝트의 하드웨어 구성에 적합합니다.

4-7) 버튼 (Button)

외부 버튼은 시스템 관리 모드로 진입하기 위해 설계된 입력 장치입니다.

- 버튼을 누르면 관리자 모드가 활성화되며, 이 상태에서 Keypad를 이용해 새로운 비밀번호를 설정할 수 있습니다.
- 버튼은 관리자만 접근 가능한 보안 강화 기능을 제공하며, 잘못된 접근을 방지하기 위해 다른 입력 장치와 구분되는 물리적 인터페이스로 설계되었습니다.
- 사용자는 버튼을 눌러 간편하게 시스템을 관리할 수 있으며, 프로젝트의 보안성과 편의성을 동시에 높이는 역할을 합니다.

5. 팀 활동 보고서

미팅 개요	일시	2024.12.05		
	참석자	홍석영		
	본 미팅의 주요활동	<div>- 개발 일정과 현재 상황을 비교하여 유동적으로 상황을 조정</div> <div>- SW 알고리즘 개발 중 발생한 문제를 팀원 및 교수님과 토의하여 해결방안을 탐구</div> <div>- 개발한 SW를 바탕으로 전반적인 하드웨어 설계</div>		
진행 사항	구분	활동 내용	조치 사항	
	이번 미팅에서 한 일	<div>- 개발 일정대로 제대로 진행되었음을 확인하여 남은 개발 일정 순차적으로 진행</div> <div>- 초음파 센서(HC-SR04)과 서보 모터 간 문제점이 발생하여 해결 방안 탐구</div> <div>- SW를 바탕으로 HW 작품 설계도 구성</div>	<div>- HW를 만들기위해 전반적인 설계도 작성 및 준비물 구매</div> <div>- 교수님의 조언을 토대로 전력 부족, 핀 충돌, 타이머 카운터 충돌 등 다양한 문제점 탐색</div>	
	논의 사항	<div>- 현재 개발 일정이 계획대로 진행되고 있는지 확인하며 남은 개발 일정에 대해 순차적 진행 계획 논의</div> <div>- 초음파 센서와 서보 모터 간 발생한 문제점을 해결하기 위하여 다양한 해결방안을 탐구하고, 이를 토대로 추가적인 테스트 및 조치사항 논의</div>		
추후 계획	구분	활동 내용	역할 분담	추진 일정
	다음 미팅에서 해야 할 일	<div>- 설계한 HW를 바탕으로 작품 만들기</div>	<div>- HW 설계</div>	2024.12.12
	기타			
피드백	교수님의 피드백 사항	<div>- 초음파 센서(HC-SR04)과 서보 모터 간 문제점은 HW 적으로는 문제가 없고, SW 알고리즘 적으로 문제가 있는걸로 판단하여 다시 코드를 짜보도록 권장</div>		

6. 수행 내용

6-1) 센서 구성

6-1-1) FND

```
// FND 관련 함수
unsigned char Port_char[] = {
    0xc0,0xf9,0xa4,0xb0,0x99,
    0x92,0x82,0xd8,0x80,0x90,
    0x88,0x83,0xc4,0xa1,0x84,
    0x8e
}; // 애노드 공통

unsigned int Port_fnd[] = {0x1f,0x2f,0x4f,0x8f}; // FND0 ON, FND1 ON, FND2 ON, FND3 ON

void PORT_Init(void) {
    DDRF = 0xf0; // FND 제어위한 출력 설정(PF4 .. 7)
    DDRB = 0xff; // FND 제어위한 출력 설정(PB0 .. 7)
    DDRC = 0xf0; // PORTC 4 ~ 7 은 Row 선택위한 출력으로 설정
    // PORTC 0 ~ 3 은 col 값 읽기위한 입력으로 설정
}

void FND_Dis(int dec) {
    PORTF = Port_fnd[0]; PORTB = Port_char[(dec / 1000 % 10)]; _delay_ms(1); PORTB = 0xFF;
    PORTF = Port_fnd[1]; PORTB = Port_char[(dec / 100 % 10)]; _delay_ms(1); PORTB = 0xFF;
    PORTF = Port_fnd[2]; PORTB = Port_char[(dec / 10 % 10)]; _delay_ms(1); PORTB = 0xFF;
    PORTF = Port_fnd[3]; PORTB = Port_char[(dec / 1 % 10)]; _delay_ms(1); PORTB = 0xFF;
    PORTF = 0x00;
}

int getNum(unsigned char keyCode) {
    unsigned char keyNum[16] = {1,2,3,'A', 4,5,6,'B', 7,8,9,'C', '*',0,'#','D'};
    if(keyCode > 16)
        return -1;
    return keyNum[keyCode - 1];
}
```

FND는 사용자에게 상태 정보를 시각적으로 제공하며, 입력된 데이터를 PORTB 및 PORTF를 통해 제어합니다.

- 구성 및 동작 원리:

- 입력 핀: PORTB 0~7번으로 값을 받아옵니다.
- 출력 핀: 상위 PORTF 핀을 사용하여 FND에 데이터를 출력합니다.
- 숫자 설정:
 - 숫자 출력을 위해 배열을 미리 생성하여 매핑합니다.
 - getNum 함수를 통해 해당 배열에서 값을 받아 FND에 출력합니다.

6-1-2) Buzzer

```
// 경고음 시작 함수
void startSiren(void) {
    TCCR1A = 0x00; // CTC 모드
    TCCR1B = (1 << WGM12) | (1 << CS11); // 분주비 8, CTC 모드
    OCR1A = 1474; // 1kHz (F_CPU / (2 * 8 * 1kHz) - 1)
    TIMSK |= (1 << OCIE1A); // 비교 일치 인터럽트 활성화
}

// 경고음 정지 함수
void stopSiren(void) {
    TCCR1B = 0x00; // 타이머 정지
    PORTG &= ~(1 << PG4); // 부저를 끄
}

// 타이머 비교 일치 인터럽트 서비스 루틴
ISR(TIMER1_COMPA_vect) {
    PORTG ^= (1 << PG4); // PG4 핀 토글 (부저 진동)
}
```

Buzzer는 청각적 피드백을 제공하며, 다양한 상황에서 경고음을 출력합니다.

- 구성 및 동작 원리:

- 연결 핀: PORTG.4번 핀과 연결됩니다.
- 제어 방식:
 - 타이머 비교 일치 인터럽트를 사용하여 다양한 음을 생성합니다.
- 출력 음:
 - 경고음: 비밀번호가 틀리거나 도난 상황 시 출력.
 - 성공음: 비밀번호 입력 성공 시 짧은 긍정적인 멜로디 출력.
 - 입력 음: 비밀번호 입력 시 짧은 확인음을 출력.

6-1-3) HC-SR04

```
void Ultrasonic_Init(void) {
    DDRE |= (1 << TRIG); // TRIG = 출력
    DDRE &= ~(1 << ECHO); // ECHO = 입력
}

// 초음파 거리 측정 함수
unsigned int measure_distance(void) {
    unsigned int distance;

    TCCR1B = 0x03; // Timer/Counter1 클럭 설정: 64분주
    PORTE &= ~(1 << TRIG); // Trig LOW
    _delay_us(10);
    PORTE |= (1 << TRIG); // Trig HIGH
    _delay_us(10);
    PORTE &= ~(1 << TRIG); // Trig LOW

    while (!(PINE & (1 << ECHO))); // Echo HIGH 대기
    TCNT1 = 0x0000; // Timer 초기화

    while (PINE & (1 << ECHO)); // Echo LOW 대기
    TCCR1B = 0x00; // Timer 정지

    distance = (unsigned int)(SOUND_VELOCITY * (TCNT1 * 4 / 2) / 1000); // mm 단위 거리 계산
    return distance;
}
```

초음파 센서는 거리 측정을 통해 물체의 유무를 감지하는 데 사용됩니다.

- 구성 및 동작 원리:

- 입력 핀: ECHO 신호는 PORTE.7번 핀으로 입력됩니다.
- 출력 핀: TRIG 신호는 PORTE.6번 핀으로 출력됩니다.

- 거리 계산 공식:
- 물체와 초음파 센서 간의 거리는 다음과 같이 계산됩니다:

$$\text{거리} = \frac{340 \times \text{TCNT1 값} \times 4}{2 \times 1000}$$

- 340은 초음파의 속도(m/s).
- TCNT1은 타이머 값을 의미합니다.

6-1-4) ServoMotor

```
// Servo_Motor 초기화
void Servo_Init(void) {
    DDRE |= (1 << PE4); // PORTE4 핀을 출력으로 설정
}

// Timer3 초기화 (서보 모터 제어용)
void Timer3_Init(void) {
    TCCR3A = (1 << WGM31) | (1 << COM3B1); // Fast PWM 모드, 비반전 출력
    TCCR3B = (1 << WGM32) | (1 << WGM33) | (1 << CS31); // 분주비 8
    ICR3 = 36864; // TOP 값 (PWM 주기 20ms, 50Hz)
    OCR3B = 3010; // 초기값 (1.5ms, 0도 위치)
}
```

ServoMotor는 보관함의 잠금장치를 제어하며, Fast PWM 모드를 활용하여 동작합니다.

- 구성 및 동작 원리:
 - 연결 핀: PORTE.4번 핀을 통해 제어 신호를 출력합니다.
 - 제어 방식:
 - Timer3을 사용하여 Fast PWM, 비반전 모드로 동작합니다.
 - OCR 값을 실험적으로 조정하여 서보 모터의 각도를 설정합니다.
 - 기능:
 - 비밀번호 입력 성공 시 잠금장치를 해제합니다.
 - 일정 시간이 지나면 다시 잠금장치를 잠급니다.

6-2) 기능 구성

6-2-1) 비밀번호 입력 및 확인

- 비밀번호 비교 함수

```
int comparePassword(unsigned char* input, unsigned char* preset, unsigned char length) {
    for (int i = 0; i < length; i++) {
        if (input[i] != preset[i]) return 0; // 비밀번호 불일치
    }
    return 1; // 비밀번호 일치
}
```

- 사용자가 입력한 비밀번호와 시스템에 저장된 비밀번호를 비교합니다.
- 모든 숫자가 일치할 경우 비밀번호가 올바른 것으로 인식합니다.
- 하나라도 불일치할 경우, 입력 오류로 처리합니다.
- 비밀번호가 맞으면 서보 모터가 작동하여 보관함이 열립니다.

6-2-2) 택배 보관 유무 확인

```
while (check_count <= 6){
    distance_mm = measure_distance(); // 거리 측정 (mm 단위)
    distance_cm = distance_mm / 10; // cm 단위로 변환
    LCD_Clear(); // LCD 화면 초기화
    sprintf(buffer, sizeof(buffer), "Distance: %dcm", distance_cm); // 거리 값 포맷
    LCD_Str((Byte*)buffer); // 거리 값 출력
    _delay_ms(500); // 0.5초 간격
    if (distance_cm < 7)
    {
        check_count += 1;
    }
    else{
        check_count = 0;
    }
}
```

- HC-SR04 초음파 센서를 이용하여 물체와의 거리를 측정합니다.
- 물체와의 거리가 7cm 이하로 유지되고, 이 상태가 3초 동안 지속되면 보관함에 택배가 보관된 것으로 간주합니다.
- 성공적인 보관 시 부저와 LCD를 통해 사용자에게 알림을 제공합니다.

6-2-3) 비밀번호 변경 모드

```
// ISR 함수
ISR(INT1_vect) {
    if (!password_change_mode) { // 비밀번호 변경 모드가 비활성화된 경우에만 동작
        password_change_mode = 1;
        state = 1; // 비밀번호 입력 활성화
        LCD_Clear();
        LCD_Pos(0, 0);
        LCD_Str("Set New Password:");
        new_char_count = 0;
        for (int i = 0; i < 10; i++) {
            new_password[i] = 10; // 새 비밀번호 초기화
        }
    }
}

// 인터럽트 초기화 함수
void interrupt_init(void) {
    EIMSK = (1 << INT1); // INT1 활성화
    EICRA = (1 << ISC11); // 하강 에지에서 트리거 (ISC11 = 1, ISC10 = 0)
    DDRD &= ~(1 << PD1); // PORTD.1을 입력으로 설정
    PORTD |= (1 << PD1); // 내부 풀업 활성화
    sei(); // 전역 인터럽트 허용
}
```

- PORTD.1번에 연결되어있는 버튼을 눌렀을 때 비밀번호 변경 모드가 활성화됩니다.
- 사용자는 새 비밀번호를 입력하고 저장할 수 있습니다.
- 외부 인터럽트를 사용하여 어떤 과정을 수행중에도 PORTD.1 번을 누르게 되면 비밀번호 변경 모드가 활성화 됩니다.

6-2-4) 청각적인 사용자 피드백

부저를 활용하여 다양한 상황에 맞는 청각적 알림음을 제공합니다.

- 비밀번호 오류 시: 짧은 경고음을 울립니다.
- 비밀번호 성공 시: 짧고 긍정적인 멜로디를 재생합니다.
- 도난 위험 발생 시: 반복적이고 긴 경고음을 울립니다.
- 비밀번호 입력시: 특정 톤으로 사용자에게 알립니다.

7. 계획 대비 달성도 평가

7-1) 달성된 기능 (구현)

보관함 잠금/해제 기능

- 키패드를 사용하여 비밀번호를 입력하면 보관함의 잠금이 해제됩니다.
- 서보 모터를 제어하여 보관함을 열거나 닫는 기능을 성공적으로 구현하였습니다.

인터페이스 기능

- LCD 및 FND 디스플레이를 통해 사용자에게 정보를 제공하고 키패드를 통해 입력을 처리할 수 있습니다.
- 사용자 피드백과 시스템 상태 표시를 효과적으로 제공합니다.

소리 알람 기능

- 부저를 활용하여 성공 또는 오류 상태를 소리로 알리는 기능이 구현되었습니다.
- 잘못된 비밀번호 입력, 도난 감지 시 경고음을 발생시킵니다.

원격지 알림 기능

- UART 통신을 통해 원격지로 알림을 전송할 수 있습니다.
- 택배 도착, 도난 경고 상황에서 원격 알림이 가능하도록 설정되었습니다.

관리자 모드 기능

- 비밀번호 변경 모드가 구현되어 관리자가 보관함 비밀번호를 수정할 수 있습니다.
- 외부 인터럽트를 통해 모드 전환을 활성화하도록 설계되었습니다.

도난 알림 기능

- 초음파 센서를 사용하여 물체 부재를 감지합니다.
- 입력이 제대로 될때 까지 도난 알림이 유지되게 합니다.

7-2) 실패한 기능(미구현)

원격 제어 기능

- 보관함을 원격으로 잠금/해제하거나 관리할 수 있는 기능은 구현되지 않았습니다.
- 이는 네트워크 모듈과 소프트웨어 개발이 추가적으로 요구되며, 향후 개발 가능성이 있는 영역으로 남아 있습니다.

8. 학습자 성찰 일지

학번	2019146037	이름	홍석영
1. 이번 학습을 통해 무엇을 배웠나요?			
ATmega128을 활용하여 GPIO, Interrupt, Timer/Counter, 직렬통신(UART) 등의 기능을 융합하여 임베디드 시스템을 개발하는 방법을 배웠습니다.			
2. 수업에서 어려웠던 활동은 무엇이었나요?			
ATmega128 간 통신(UART, TWI)을 배우고 이를 프로젝트에 적용하는 과정에서, 처음 접하는 실습이 다 보니 통신 설계와 구현 과정에서 어려움을 겪었습니다.			
3. 앞으로 내가 더 알고 싶은 내용은 무엇인가요?			
ATmega128 간 통신 이외에도 Arduino나 Raspberry Pi와 같은 다양한 보드 간의 통신 방법을 배우고 싶습니다. 또한 프로젝트에 사용한 센서 외에 다른 센서의 데이터시트를 분석하고 활용 방법을 익히고자 합니다.			
4. 학습한 내용을 적용할 수 있는 것은 무엇인가요?			
IoT 통신을 활용하여 임베디드 보드에서 수집한 센서 데이터를 다른 임베디드 보드와 통신 제어에 활용할 수 있으며 임베디드 시스템을 활용한 다양한 프로젝트 설계에 적용할 수 있습니다.			
5. 이 수업에서 나의 부족한 부분은 무엇인가요?			
임베디드 보드 간 통신(UART, I2C, TWI)에 대한 지식이 부족하여 통신 이론에 대한 이해와 설계 역량이 부족함을 느꼈습니다.			
6. 이 수업의 학습과정을 통해 무엇을 느꼈나요?			
임베디드 개발에서 알고리즘 설계와 소프트웨어 역량뿐만 아니라, 센서와 GPIO를 연결하고 하드웨어를 설계하는 과정도 매우 중요하다는 것을 깨달았습니다.			
7. 기타 느낀 점을 자유롭게 기술하세요.			
임베디드 보드를 실제로 활용할 기회가 많지 않았는데, 이번 프로젝트를 통해 임베디드 보드로 개발을 직접 경험하고, 만들고 싶었던 제품을 구현할 수 있어서 큰 만족감을 얻었습니다. 이러한 경험이 임베디드 개발자로서의 자신감을 높여주었고, 앞으로 더 다양한 프로젝트를 통해 역량을 강화하고 싶다는 동기를 얻었습니다.			

9. 맺음말

이번 ‘도난 방지 택배 보관함’ 프로젝트를 통해 ATmega128의 GPIO 제어, 센서 간 인터페이스, UART 통신, 하드웨어 설계 등 이론으로만 배웠던 내용을 실습을 통해 구현하며 임베디드 개발자로서의 역량을 키울 수 있었습니다. 이 과정에서 실제 시스템 설계와 디버깅을 경험하며 실무에 필요한 문제 해결 능력도 향상되었습니다.

다만 MCU 간 통신을 처음 다루다 보니 UART를 이용한 메시지 전송 기능에만 그쳤고, 다른 통신 방식이나 추가 기능을 구현하지 못한 점이 아쉬움으로 남습니다. 만약 다양한 통신 방식과 추가 기능 (예: IoT 연동, 보안 강화)을 구현했다면 프로젝트의 완성도가 더 높아졌을 것으로 생각합니다.

이번 경험을 바탕으로 향후 프로젝트에서는 보다 복합적인 기능과 고도화된 시스템을 설계할 수 있도록 지속적으로 역량을 개발해 나가겠습니다.