**Project**

**‘Good quality factory’**

****

팀 이름: N.S.F(Non Stop Factory)

장유정

E-MAIL : soclsrndirh@gmail.com

Github : https://github.com/jjang-yu

백승기

E-MAIL : back2back2@naver.com

Github : https://github.com/seunggi-baek

강시은

E-MAIL : kse3848@gmail.com

Github : https://github.com/sieun-20

김희철

E-MAIL : kkhc1028@naver.com

한국직업능력교육원 안산

**1.개요**

1.1 프로젝트 명

1. "Good quality factory"라는 프로젝트명은 품질에 대한 우수성과 공장 제조과정의 효율성을 강조하기 위해 선택되었습니다. 이 프로젝트는 제품이나 서비스의 품질 향상에 중점을 두며, 공장 내의 생산 및 관리 과정을 개선하여 더 나은 제품을 생산하는 것을 목표로 합니다. 고객들에게 더 나은 제품을 제공하고, 동시에 공장 내부의 생산 프로세스를 최적화하여 비용과 시간을 절약하는 것을 목표로 합니다.

1.2 추진 배경

- 비용절감 및 효율성 향상: 실시간으로 재고를 모니터링하고 관리함으로써 과다한 재고를 줄이고, 재고 부족으로 인한 물류 문제를 방지할 수 있으며, 이를 통해 재고 관리 비용을 절감하고, 생산 및 유통 과정에서의 효율성을 높일 수 있다.

- 고객 서비스 향상: 실시간으로 재고 상태를 파악하고 관리함으로써 고객 주문에 대한 신속한 대응이 가능해져서 고객 만족도를 높일 수 있으며, 재고 부족이나 오류로 인한 주문 지연을 방지하여 고객들에게 신속하고 정확한 서비스를 제공할 수 있다.

- 정확한 데이터 기반의 의사 결정: 실시간으로 모니터링되는 데이터를 기반으로 재고 관리 및 물류 프로세스를 최적화할 수 있으며, 이를 통해 데이터 기반의 의사 결정을 통해 비즈니스 운영을 최적화하고 경쟁력을 확보할 수 있다.

- 회계 및 감사 효율성 증대: 입출고 기록을 실시간으로 관리함으로써 회계 및 감사 과정에서의 효율성을 높일 수 있으며, 정확하고 신속한 입출고 기록은 재무 상태 및 재고 가치를 정확하게 파악할 수 있도록 도와준다.

- 법규 준수 및 안전 보장: 실시간으로 재고 및 입출고를 모니터링하고 기록함으로써 규제 요구 사항을 준수하고 안전한 물류 프로세스를 보장할 수 있다. 예를 들어, 유통 기한이 지난 제품의 처리나 특정 제품의 안전한 보관 등에 대한 요구 사항을 충족할 수 있다.

- 작업 안전성 향상

2. 프로젝트 범위

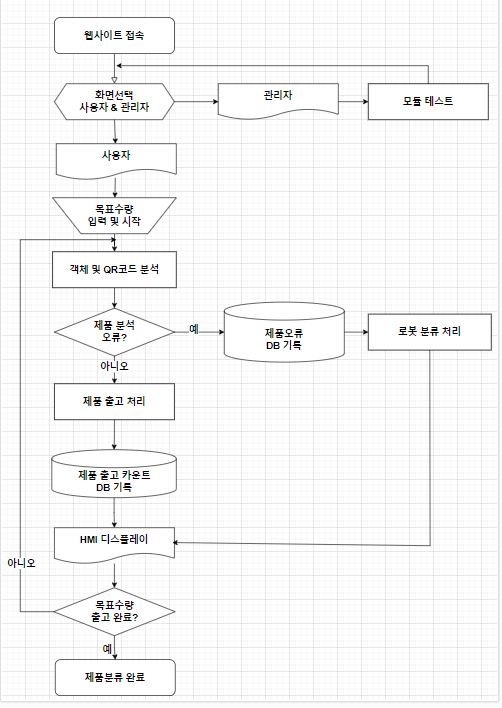
2.1 작업명세서

|  |  |
| --- | --- |
| 업무 | 업무범위 |
| 라즈베리파이 (서버) | 1. 웹서버/사이트 구축 (Apache2, Flask, MariaDB, WebSocket)  2. 라즈베리파이 (객체감지) 연동 (ssh 원격 텔넷)  3. 아두이노 (마스터) 연동 (logic level converter 이용 uart 통신)  4. 타워램프 제어 |
| 라즈베리파이 (객체감지) | 1. 웹캠 이미지, 색상 오류 기능 개발 (Yolov5)  2. 파이캠 외형, QR코드 오류 기능 개발 (OpenCV) |
| 아두이노 (마스터) | 1. 컨베이어 스텝모터 제어  2. 적외선 및 스위치 센서 제어  3. 아두이노 (슬레이브) 연동 (I2C 통신) |
| 아두이노 (슬레이브) | 1. 로봇팔 내부 서보모터 제어  2. 제품분류 서보모터 제어  3. 아두이노 (마스터) 연동 (I2C 통신) |
| 문서화 | 간트 차트, 플로우 차트, 회로도 |

- 간트 차트

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **업무 내용** | **담당자** | **2024년 3월** | | | | **4월** | | | |
| **2주 8~10** | **3주 11~17** | **4주 18~24** | **5주 25~31** | **1주 1~7** | **2주 8~14** | **3주 15~21** | **4주 22** |
| 서버 셋팅 | 백승기 |  |  |  |  |  |  |  |  |
| 웹서버/사이트 구축 | 백승기 |  |  |  |  |  |  |  |  |
| 라즈베리파이 (객체감지) 연동 | 백승기 |  |  |  |  |  |  |  |  |
| 타워램프 제어 | 백승기 |  |  |  |  |  |  |  |  |
| 웹캠 이미지, 색상 감지 기능 개발 | 장유정 |  |  |  |  |  |  |  |  |
| 파이캠 외형, QR 감지 기능 개발 | 장유정 |  |  |  |  |  |  |  |  |
| 화면 개발 | 백승기 |  |  |  |  |  |  |  |  |
| 이미지 및 색상 오류 개발 | 장유정 |  |  |  |  |  |  |  |  |
| 외형 및 QR 오류 개발 | 장유정 |  |  |  |  |  |  |  |  |
| 컨베이어 스텝모터 제어 | 강시은 |  |  |  |  |  |  |  |  |
| 물품 선별 기능 구현 | 강시은, 김희철 |  |  |  |  |  |  |  |  |
| 적외선 및 스위치 센서 제어 | 강시은 |  |  |  |  |  |  |  |  |
| 로봇팔 제작 및 기능 구현 | 김희철, 장유정 |  |  |  |  |  |  |  |  |
| 제품분류 서보모터 제어 | 김희철 |  |  |  |  |  |  |  |  |
| 아두이노 연동 (I2C통신) | 강시은, 김희철 |  |  |  |  |  |  |  |  |
| 아두이노 연동(uart통신) | 강시은, 김희철 |  |  |  |  |  |  |  |  |
| 회로도 제작 (회로, PCB) | 강시은 |  |  |  |  |  |  |  |  |
| 전체 Layout 작성 및 배치 | 강시은, 김희철 |  |  |  |  |  |  |  |  |
| 공정 연동 및 테스트 | 팀원 전체 |  |  |  |  |  |  |  |  |

- 플로우 차트



3. 시스템 구축 환경

3.1 하드웨어 구축

- 하드웨어 구성 현황

|  |  |  |
| --- | --- | --- |
| **하드웨어** | **모델명** | **상세설명** |
| 라즈베리파이4 1 | 라즈베리파이4 1 | 웹서버, 데이터베이스 구축 |
| 라즈베리파이4 2 | 라즈베리파이4 2 | 객체감지 |
| LCD | Raspberry-Pi Touch Display  (Rev 1.1) | 웹서버 운영 |
| Arduino | Arduino UNO | 아두이노간 통신, 서보모터 제어 |
| 파이캠 | APC480 | QR 및 외형 감지 |
| 웹캠 | 라즈베리파이 카메라모듈 V2, 8MP | 이미지 및 색상 감지 |
| 컨베이어 | 싸이피아 (A1)미니 탁상용 컨베이어벨트 | 제품 이동 |
| 로봇팔 | 코코아팹 로봇팔 키트 | 제품 오류 분류 |
| 서보모터 | SG90 x 4 | 제품 분류 |
| 타워램프 | ST45L LED | 공정 상태 표시 |

1. 소프트웨어 개발 언어 및 IDE 구성현황

|  |  |  |
| --- | --- | --- |
| **언어** | **IDE** | **구성내용** |
| C/C++ | Arduino IDE | 컨베이어 스텝모터, 로봇팔 서보모터, 제품분류 서보모터 제어 코드 작성 스위치, 적외선 센서 제어 코드 작성 라즈베리파이 (서버) 와의 시리얼 통신 및 아두이노 (마스터/슬레이브) 시리얼 통신 코드 작성 |
| Python3 | Thonny | 라즈베리파이 (객체감지) 와의 원격 ssh 통신 코드 작성 아두이노 (마스터) 와의 시리얼 통신 코드 작성 Flask 서버 구축 및 웹소켓을 이용한 각 페이지별 통신 코드 작성 OpenCV, Pytorch, Tensorflow, Yolov5객체 인식 코드 작성 라즈베리파이(서버)를 통하여 데이터베이스에 데이터 전송 |
| MariaDB | Workbench | 서버구축 및 데이터베이스 작성 및 관리 |

4. 프로젝트 추진체계

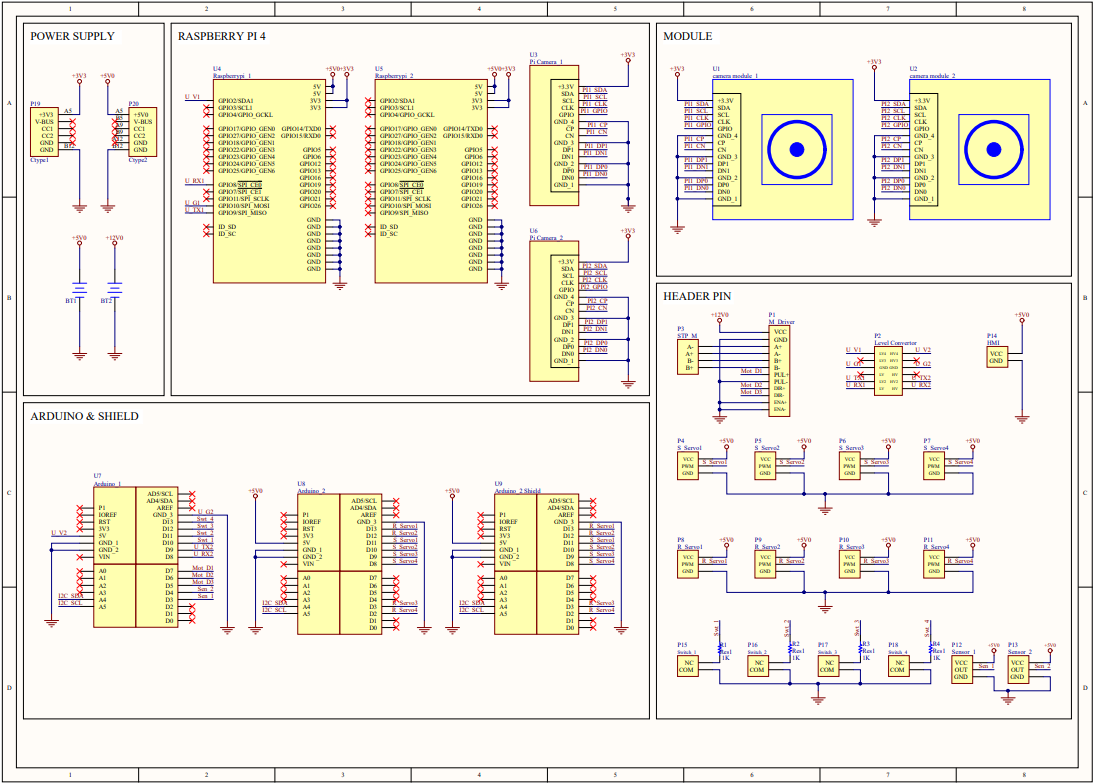
4.1 조직별 역할

|  |  |
| --- | --- |
| 성명 | 직무 |
| 장유정 | . 웹캠 이미지, 색상 오류 기능 개발 (Yolov5)   1. 파이캠 외형, QR코드 오류 기능 개발 (OpenCV) 2. 이미지 촬영, 라벨링 (Roboflow) 3. Dataset을 이용한 학습 파일(.pt) 생성 |
| 강시은 | 1. 회로도 제작 (Circuit Design, PCB Artwork) 2. 기판 제작 (회로 설계 및 정리) 3. 컨베이어 스텝모터 제어 4. 분류 스위치 제어 5. 아두이노 (마스터) I2C 통신 설계 6. 아두이노 (마스터) uart 통신 설계 |
| 김희철 | 1. 로봇팔 서보모터 제어 2. 분류 서보모터 제어 3. 순서도(Flowchart) 구조 설계 4. 아두이노 (슬레이브) I2C 통신 설계 5. 아두이노 (마스터) uart 통신 설계 |
| 백승기 | - 웹서버/사이트 구축 (Apache2, Flask, MariaDB, WebSocket)  - 라즈베리파이 (객체감지) 연동 (ssh 원격 텔넷)  - 아두이노 (마스터) 연동 (logic level converter 이용 uart 통신)  - 타워램프 제어 |

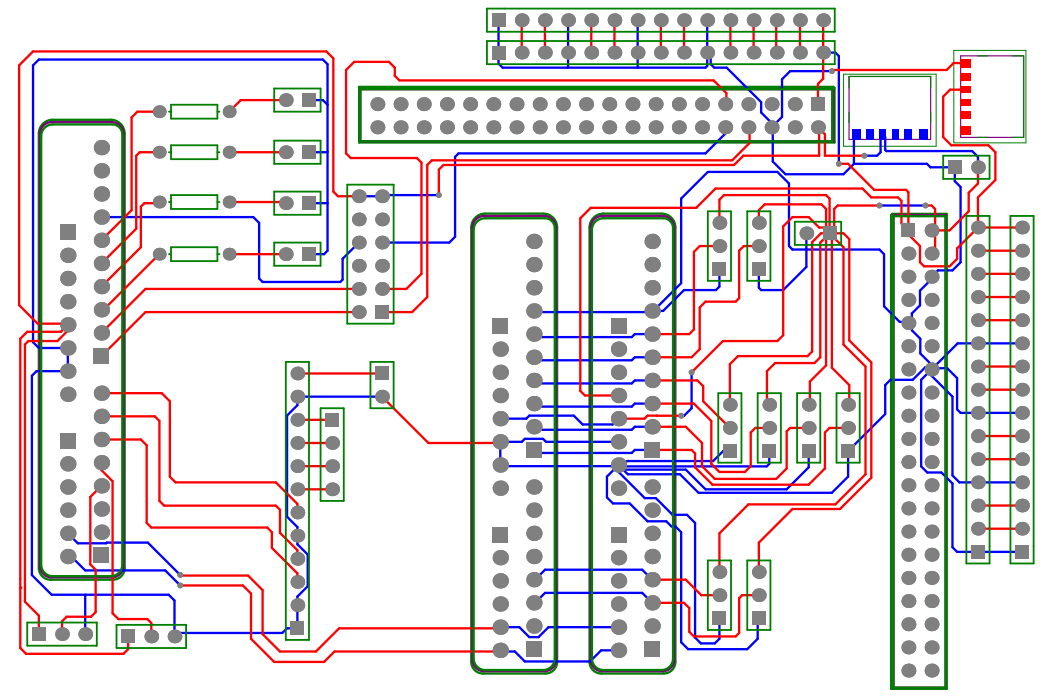
5. 프로젝트 관리

5.1 프로젝트 세부 내용

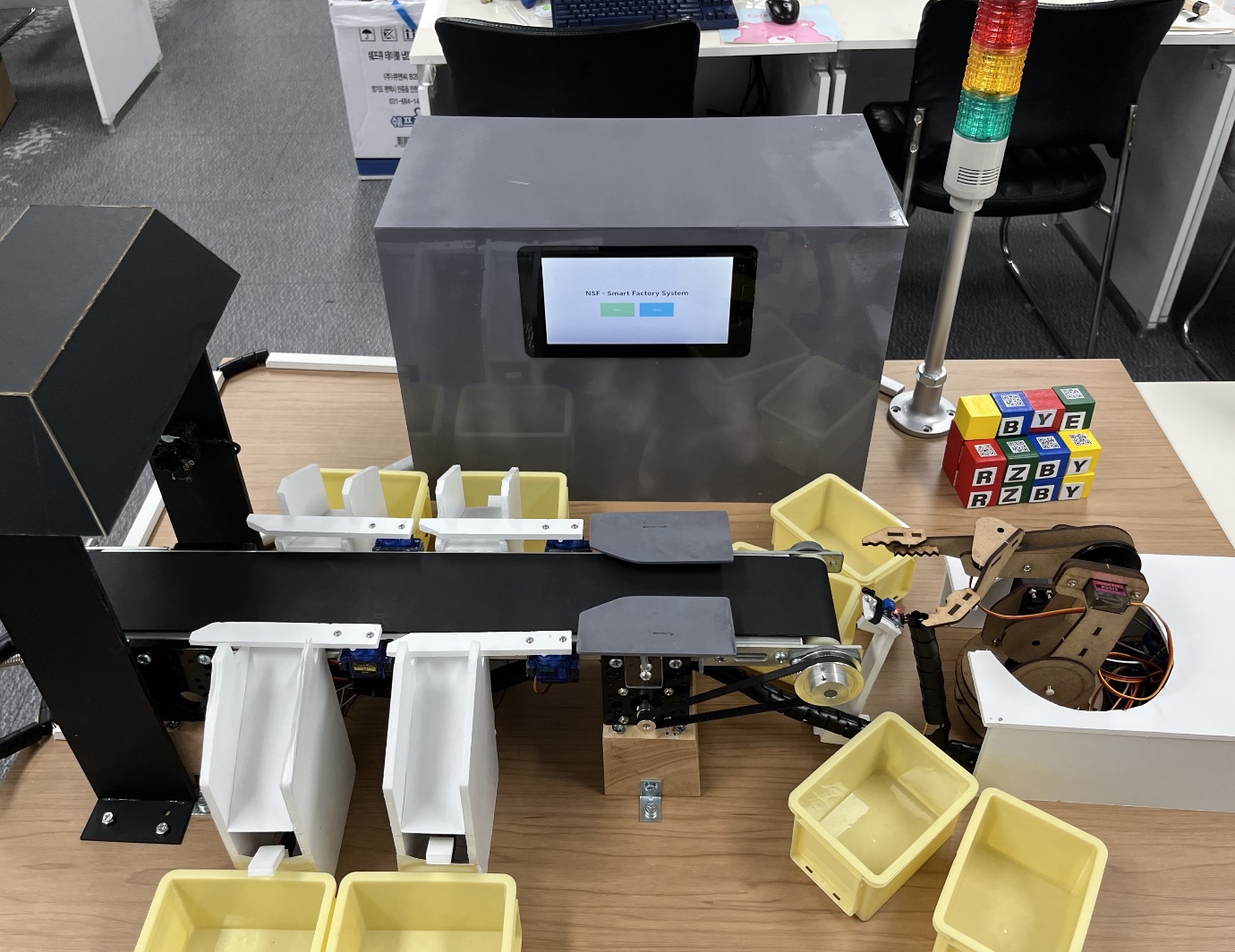
- 회로도



1. PCB



1. H/W 외형



81. 로봇팔 내부 서보모터 제어

2. 제품분류 서보모터 제어

**3. 아두이노 (서버) 연동 (I2C 통신)**

71. 로봇팔 내부 서보모터 제어

2. 제품분류 서보모터 제어

**3. 아두이노 (서버) 연동 (I2C 통신)**

41. 로봇팔 내부 서보모터 제어

2. 제품분류 서보모터 제어

**3. 아두이노 (서버) 연동 (I2C 통신)**

51. 로봇팔 내부 서보모터 제어

2. 제품분류 서보모터 제어

**3. 아두이노 (서버) 연동 (I2C 통신)**

31. 로봇팔 내부 서보모터 제어

2. 제품분류 서보모터 제어

**3. 아두이노 (서버) 연동 (I2C 통신)**

61. 로봇팔 내부 서보모터 제어

2. 제품분류 서보모터 제어

**3. 아두이노 (서버) 연동 (I2C 통신)**

**1**

21. 로봇팔 내부 서보모터 제어

2. 제품분류 서보모터 제어

**3. 아두이노 (서버) 연동 (I2C 통신)**

|  |  |
| --- | --- |
| **1-2**  **1-1** | 1-1) 파이캠  1-2) 웹캠  2) 제품 분류 서보모터  3) 컨베이어 스텝모터  4) 적외선 센서  5) 로봇팔 서보모터  6) 제품 카운트 스위치  7) 타워램프  8) HMI Display |

6. 주의사항 및 프로젝트 후기

6.1 전원 관련 주의사항

- 컨베이어 스텝모터, 타워램프의 외부 전압이 12V 이상 입력되어야 함.

- 스텝모터 발열이 심해 오래 연결하지 말 것.

6.2 카메라 주의사항

- 제품 분류 중 라즈베리파이 (객체감지)에서 카메라 테스트를 동시에 진행하지 말 것.

- 조명과 거리에 따른 객체감지가 원활하지 않을 수 있음.

- QR코드 인식 시, 800만 화소 이하의 카메라는 인식이 힘듦.

6.3 라즈베리파이 (서버) 주의사항

- 아두이노 (마스터)와 통신 시, 아두이노 (마스터/슬레이브) 통신이 원활하지 않을 경우 동작이 멈춤. (아두이노 슬레이브에서 리턴 데이터를 꼭 확인)

6.4 아두이노 (마스터) 주의사항

- 라즈베리파이 (서버)와 시리얼 통신 시, 처음엔 단방향 컨버터를 사용하였으나 통신이 되지 않아 양방향 컨버터 (로직레벨변환기 4채널 이용)를 사용함.

6.5 로봇팔 주의사항

- 전력 공급이 불안정할 경우, 아두이노 (마스터)에 리턴 데이터를 주지 못함.

6.6 이슈

- 기판에 로직레벨변환기를 납땜이 잘 되지 않아, 통신이 원활하지 않음. (배선을 통해 연결함.)

- 스위치의 GND 연결이 되지 않을 경우, 통신에 문제가 발생함.

6.7 후기(문제점 및 개선 방안)

- 라즈베리파이 (서버)에서 아두이노 (마스터)로 시리얼 통신 시, 아두이노 (슬레이브)에서 데이터를 받지 못하면 공정이 멈춰버리는 문제가 발생. 이 부분은 멀티 스레드를 이용하는 방법으로 개선이 필요함.

- 외형 검출 시, 인식할 객체의 면적 조정을 하지 않으면 다른 객체들도 무작위로 검출하게 되어 조정이 필요함.

- 구매 컨베이어 벨트 색상이 초록으로 초록 객체와 같이 인식되어버리는 문제가 발생하여, 컨베이어 벨트의 색상을 무광 검정으로 도색함.

- 객체 인식을 할 때 알 수 없는 새로운 변수들이 발생하여, 많은 이미지 학습이 필요함.

7. 참조

7.1 시연 영상

- <https://www.youtube.com/watch?v=jdg1TPM2qg0>

7.2 참고자료

- 김성우 저(2022) 사물인터넷을 품은 라즈베리 파이(개정판): 사물인터넷 프로그래밍의 모든 것

- 김성엽 저(2017) Do it! C 언어 입문

- 박응용 저(2021) Do it! 점프 투 플라스크: 파이썬 웹 개발부터 배포까지!

- 정현희 저(2023) 알기쉬운 파이썬 SQL 코딩하기