

IoT 설계 예비 보고서

식당에서 일하는 모든 알바생의 대타를 뛰어줄

똑똑한 서빙 로봇, Dae-Ta!

2021년 11월 23일

한동대학교 전산전자공학부

21900030 공혜정

21800409 신지영

(1) 소개

이번 IoT 과제주제로 우리가 제시하는 로봇은 식당에서 일하는 모든 알바생을 대신하여 훌 서빙을 도와줄 서빙 로봇이다.

로봇이 주방 입구에 있을 때에는 관리자가 테이블을 입력하여 서빙 명령을 지시할 수 있다. 로봇이 서빙을 지시받은 후에는 해당 테이블로 이동하여 인사 메세지와 함께 손님에게 음식을 받으라는 안내 문구를 출력한다. 서빙하는 중간에 어떤 테이블에서 벨을 누르면 모든 서빙이 마무리 될 때까지 대기하고 있다가, 관리자가 입력한 모든 서빙이 끝난 후 해당 테이블로 이동하여 주문과 결제를 받는다. 또한 이동 중 누군가가 로봇을 건드리면 멈추도록 동작한다.

다음과 같이 Raspberry Pi1(상단부)에서는 로봇의 **User interface** 스크린과 GUI를 통한 **Input, Output**을 맡고 있으며 Raspberry Pi2(하단부)에서는 로봇의 Actuator 부분인 **바퀴 모터와 소리 Output, 무게 센서, 카드리더기, 상태 LED Output** 등을 맡고 있다.

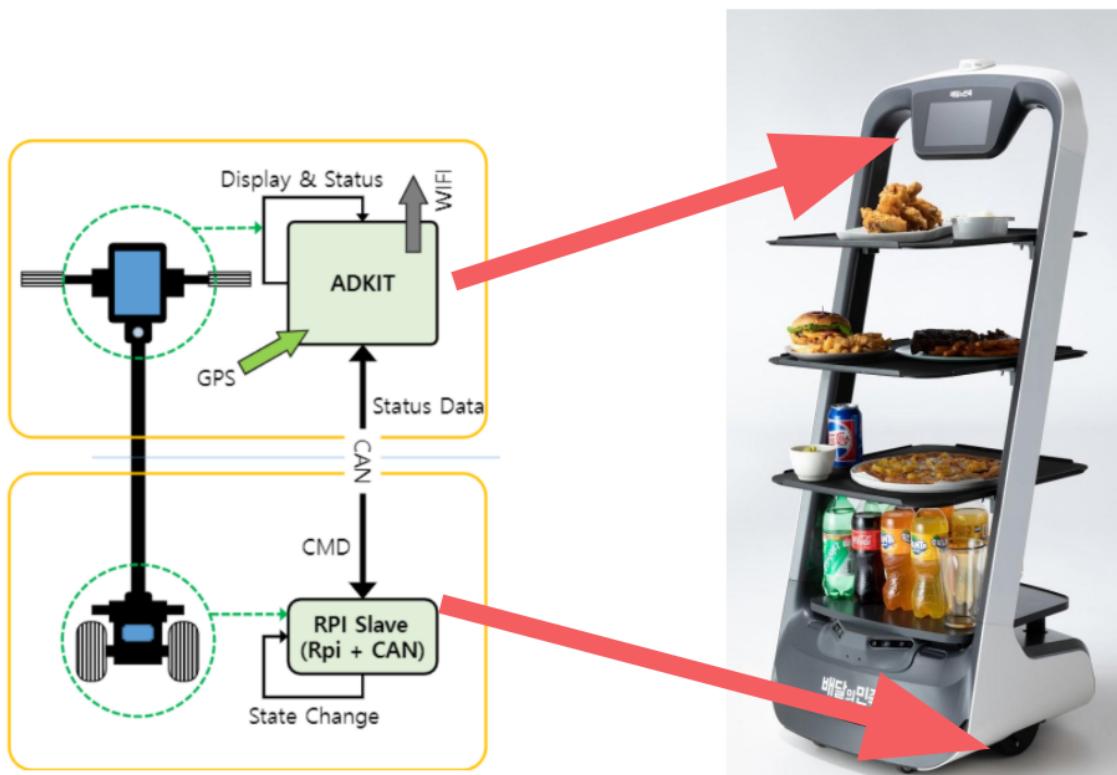


Figure 1. Raspberry Pi1과 Raspberry Pi2의 역할 모식도

(2) 상세 기능

서빙 지시 받기

Raspberry Pi1(상단부)에서는 UI를 통해 관리자의 서빙 지시를 받을 수 있다. 이때 서빙하려 이동할 테이블의 번호와 순서를 옵션으로 입력받고 서빙 출발을 입력받는다.

이동

Raspberry Pi1(상단부)에서 관리자 지시 또는 고객이 누른 벨에 의해 Raspberry Pi2(하단부)에게 이동 명령을 내린다. 그렇게 되면 Raspberry Pi2(하단부)에서는 Raspberry Pi1(상단부)에서 지시한 테이블로 이동한다. 이 때, 속도와 방향을 제어 할 수 있어야하고, 각 테이블로 이동할 경로를 Raspberry Pi2(하단부)는 기억하고 수행해야한다. 이동 중 장애물(방해)이 있을 시에 Raspberry Pi2(하단부)에서 인식을 하고, 상태LED를 주황빛으로 점등하고, 이동을 잠시 멈춘다.

인사

지시된 테이블로 이동이 끝난 후(도착한 후)에는 로봇은 스피커와 스크린을 통해 인사를 한다. 이때 로봇(Raspberry Pi1)은 현재 상황이 음식을 서빙하러 온 상황인지, 주문을 받으러 온 상황인지 state변수를 통해 판단한다.

주문과 결제

고객은 스크린(Raspberry Pi1,상단부)을 통해 주문을 하고, 카드 리더기(Raspberry Pi2, 상단부)를 통해 결제를 수행한다. 고객이 로봇에서 음식을 가져 갔는지를 로봇의 무게센서(Raspberry Pi2,하단부)로 감지를 하여 판단하고, 수행을 진행한다.

상태

로봇은 자신의 상태를 LED(Raspberry Pi2,하단부)로 표현한다. 자신의 위치, 이동, 결제, 배터리를 각 LED 5, 6, 14, 11/12로 점등을 통해 상태를 알린다. 또한, LED 13(RGB_LED)를 통해 로봇이 쉬는 중인지, 주문을 받는 중인지, 음식을 이동중인지, 방해를 받는지 등을 색깔로 나타낸다.

(3) UI



Figure 2. Raspberry Pi1에서의 GUI 구성

식당 안에 테이블이 총 6개 있으며 다음과 같이 식당 맵이 구성되었다고 할 때, Raspberry Pi1(상단부)에서 디스플레이 되는 화면의 첫 부분은 위와 같이 8개의 버튼과 2개의 테이블로 구성된다.

먼저 로봇의 위치인 x,y가 원으로 맵 안에 표시되며, 기계의 위치에 따라 상태 stateLocation은 세가지 상태를 갖는다.

- stateLocation="home"
- stateLocation="tableN"
- stateLocation="moving"

첫번째 상태인 “home”은 로봇이 주방 앞에 있을 때이며, 관리자가 btnTableN을 클릭하여 tableServingOrder 테이블에 서빙지시순서를 입력할 수 있다. 입력이 끝난 후 갖다주기 버튼을 누르게 되며 이때는 btnOrderOrServe가 서빙지시버튼으로 활성화된다.

두번째 상태인 “tableN”은 로봇이 N번째 table에 있는 상태이며, 손님이 음식을 받거나 주문하기 버튼을 눌러 주문 또는 결제를 할 수 있다. btnOrderOrServe를 뺀 나머지 버튼은 비활성화 된다.

세번째 상태인 “moving”은 로봇이 home도 tableN도 아닌 움직이고 있는 상태를 의미하며 이때는 모든 버튼이 비활성화된다.

로봇이 서빙지시를 받거나, 서빙이동중이거나, 도착해서 안내메세지를 출력중이거나, 또는 도착해서 주문 또는 결제를 받고 있을때 다른 테이블에서 Bell을 누르게 되면 tableBellOrder에 벨을 누른 테이블이 예약된다. 그리고 모든 해야할 일을 마친 로봇은 그 Bell을 누른 table로 순서대로 이동하게 된다.

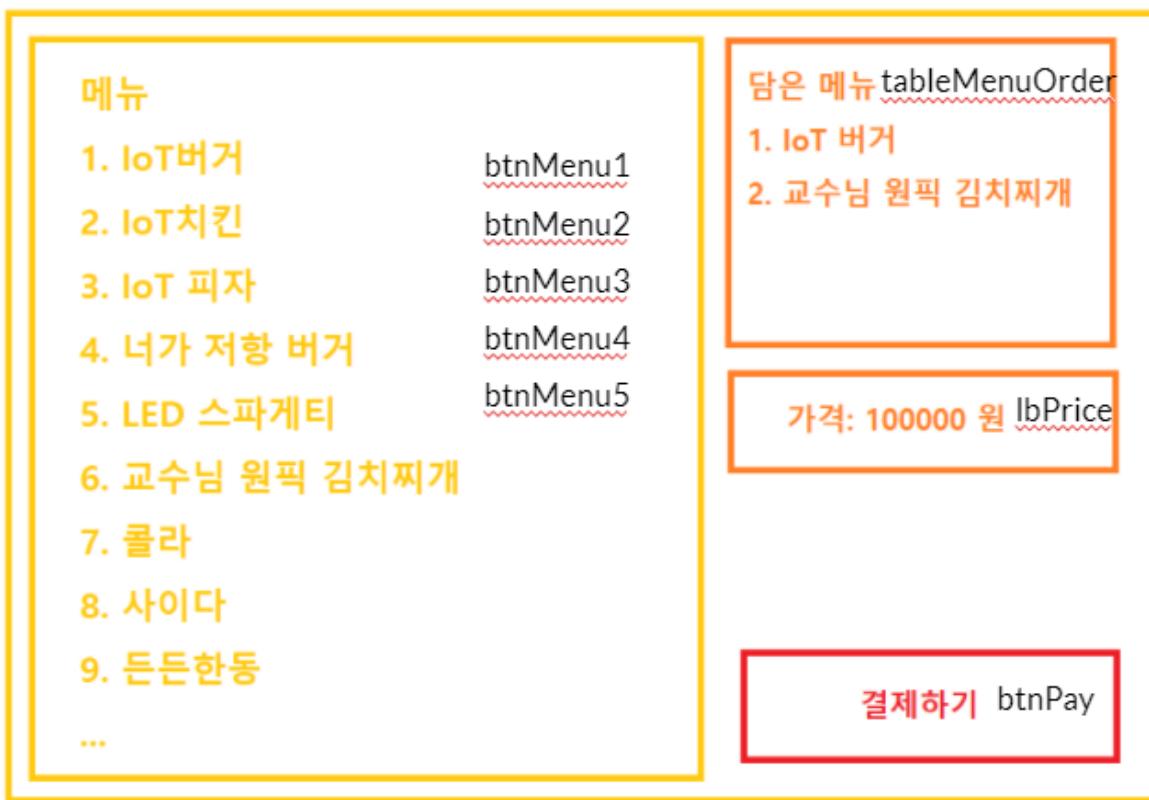


Figure 3. Raspberry Pi 1에서의 GUI 구성 2 (주문하기 메뉴)

state.location="tableN"인 상태에서 손님이 btnOrderOrServe를 누르게 되면 주문한다는 뜻으로 알고 위의 화면으로 넘어간다. 왼쪽에는 btnMenuN의 리스트가 있어서 주문 가능한 메뉴들의 이름과 가격을 보여준다. 원하는 메뉴를 골라 btnMenuN을 누르면 tableMenuOrder에 담기며 tableMenuOrder에 있는 메뉴들의 가격이 합산되어 lbPrice에 출력된다. 손님은 이를 모두 확인 후 btnPay를 누르면 결제하기 상태가 되어 카드 결제를 받는다. 카드는 편의상 꽂혀있어 결제가 항상 성공하는 것으로 간주하고 주문 성공했다는 메시지를 띄운다.

(4) 통신 방법: Socket 통신

Dae-Ta는 Socket 통신을 통해 Raspberry Pi1(상단부)와 Raspberry Pi2(하단부)가 데이터를 주고받는다. 원래의 설계 과제는 CAN통신을 이용한 프로젝트였지만, 본 프로젝트에서는 CAN통신보다 Socket통신을 사용하는 것이 더 적절한 방향이고, CAN통신에서도 Socket을 사용하기에 Socket통신을 사용하여 프로젝트를 수행한다.

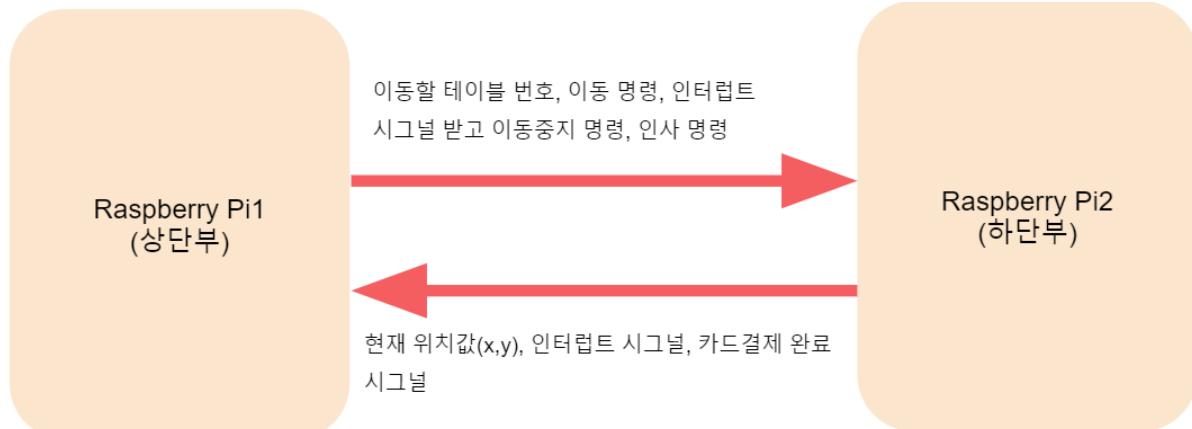


Figure 4. Raspberry Pi1과 Raspberry Pi2의 소켓 통신 모식도

Raspberry Pi1 (상단부)->Raspberry Pi2 (하단부) 데이터 송신

이동 제어 명령

```
data = {
    activity: int // 1(go), 0(stop), -1(interrupted)
    tableNo: int[], // 1(go)일 때 이동할 테이블 순서
}
```

인사 명령

```
data = {
    serviceMode: int, // 1(service), 0(nothing) // 음식이 도착한 상황
    orderMode: int // 1(orderMode), 0(nothing) // 주문받으러 도착한 상황
}
```

Raspberry Pi2 (하단부)->Raspberry Pi1 (상단부) 데이터 송신

현재 위치값(x,y), 물건 내림 시그널, 인터럽트 시그널, 카드결제 완료 시그널

현재 위치 보고

```
data = {  
    x: int,  
    y: int  
}
```

물건 내림 시그널

```
data = {  
    empty: int, // 1(empty), 0(still left)  
}
```

인터럽트 시그널

```
data = {  
    interrupted: int, // 1(interrupted), 0(okay)  
}
```

(5) Hardware Mapping in Raspberry Pi 2

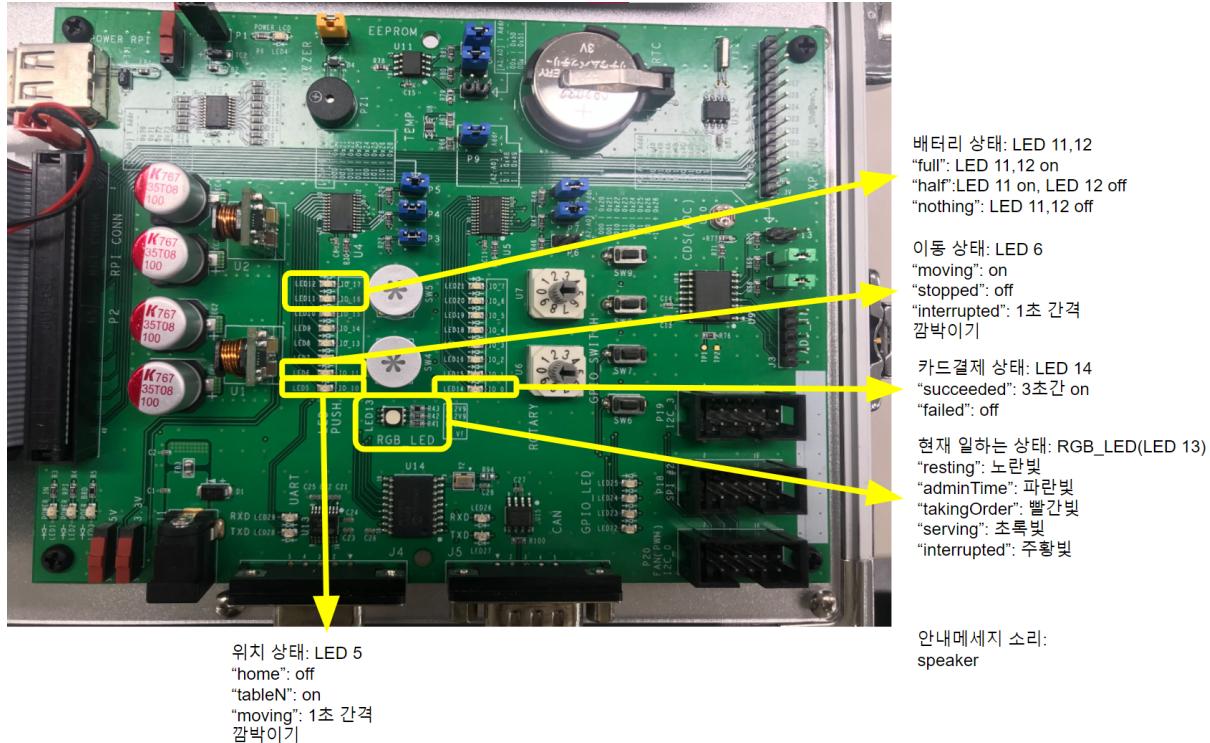


Figure 5. Raspberry Pi2의 Hardware Mapping

위치 상태: LED 5

- "home": off
- "tableN": on
- "moving": 1초 간격
깜박이기

이동 상태: LED 6

- "moving": on
- "stopped": off
- "interrupted": 1초 간격
깜박이기

카드결제 상태: LED 14

- "succeeded": 3초간 on
- "failed": off

현재 일하는 상태: RGB_LED(LED 13)

- “resting”: 노란빛
- “adminTime”: 파란빛
- “takingOrder”: 빨간빛
- “serving”: 초록빛
- “interrupted”: 주황빛

안내메세지 소리: speaker (3.5mm 단자 AUX 또는 bluetooth 연결)

배터리 상태: LED 11,12

- “full”: LED 11,12 on
- “half”: LED 11 on, LED 12 off
- “nothing”: LED 11,12 off

(6) 결론

결론적으로 서빙을 하는 직원을 보조하고, 돋는 서빙로봇(Dae-Ta)를 시스템 가상 제어 및 통합 제어판 설계하는 프로젝트를 진행한다. 필요한 곳의 요청에 의한 과제 수행을 2개의 Raspberry Pi로 가상 제어를 하고, Socket통신을 통해 데이터를 주고받는 것을 구현하고, QT로 통합 제어판 설계한다.