

2019 전기 졸업과제

중간보고서

지도교수: 김정구교수님
23조 [개냥이] 팀

201469118 이규원

201624428 김범미

201612148 전양희

1. 과제 목표

현대 사회의 바쁜 현대인들에게, 애완동물을 기를 만한 시간과 형편이 부족하다 그러나 동물을 기르고 싶은 사람들에게 조금이나마 만족감을 주기 위해서 로봇 펫을 개발한다. 개발 모델은 소니의 아이보 및 여러 로봇 펫들을 참고하였다. 이 인공지능 로봇 '쿠키'는 기본적으로 아두이노 RC카 모듈을 베이스로 하고 있으며, 블루투스과 음성 인식 오픈소스를 기반으로, 이름을 부르면 반응을 하고, 간단한 명령을 수행할 수 있도록 실제 강아지의 행동과 유사하게 만들 예정이다.

2. 과제 요구사항과 제약 사항

2.1 요구조건 분석 및 진행/수정

- 로봇 펫을 조작할 수 있는 어플리케이션 설계

→ 기본적으로 스마트폰을 통해 로봇 펫과 교감하므로, 현재 블루투스과 음성인식 기능을 탑재한 어플리케이션을 제작 중이다.

- 사용자의 명령어를 인식하는 음성 인식 기능 구현

→ 구글의 Google-Cloud-Speech와 Speech-to-Text api를 이용할 계획이었으나, Kakao의 음성인식 api뉴튼이 Google의 api보다 비용적인 측면에서 우수하고, 성능도 별 다른 차이가 없다고 판단하여, 음성인식 api를 변경하게 되었다.

- 명령어를 전달하는 블루투스 기능 구현

→ 어플리케이션에 탑재할 블루투스 연결 기능은 현재 테스트 완료 된 상태이며, 기능에 통합 될 예정이다.

- 로봇 펫의 하드웨어 구축

→ 로봇 펫의 하드웨어는 아두이노 RC카를 베이스로 제작하고, 귀와 꼬리는 추가적으로 제작할 예정이다.

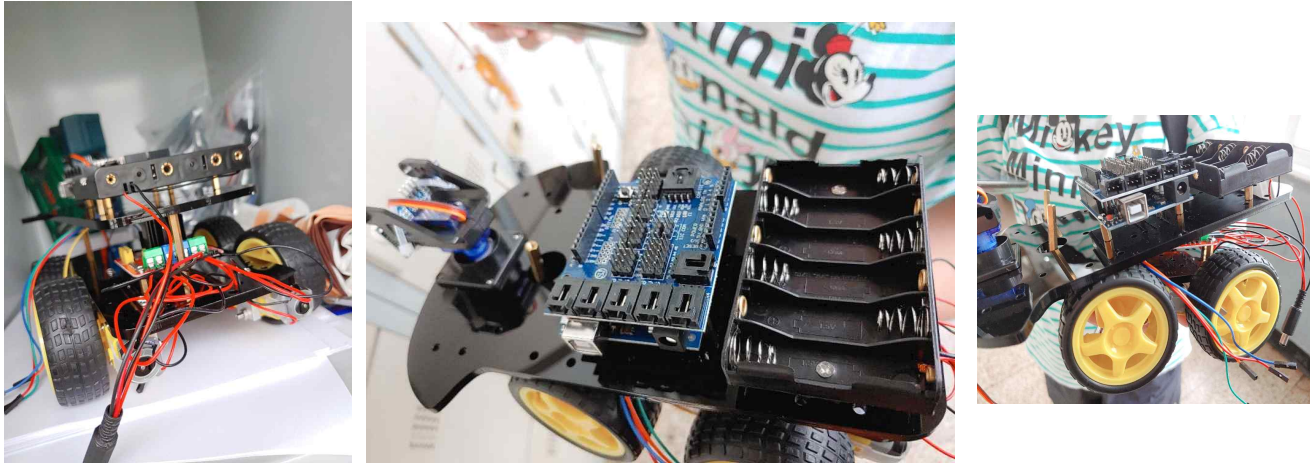
2.2 제약 사항

- 로봇 펫의 이동을 사족보행으로 하기는 비용과 노력이 지나치게 많이 든다.

→ 따라서 불필요한 노력을 줄이기 위해, 이동은 RC카를 베이스로 하여 바퀴로 이동한다.

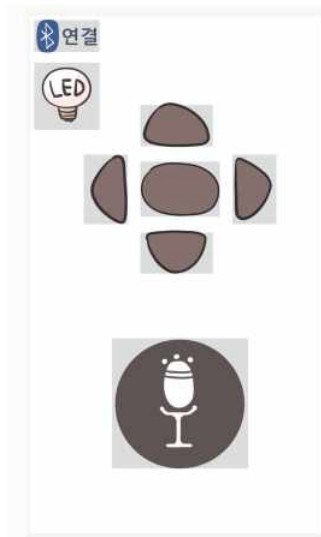
3.2 조원별 진행상황 및 앞으로 개발할 내용

● 공통 진행 상황



로봇 펫 하드웨어 조립

● 이규원



```
RCcar1
//자동차 움직임 함수들
void forward() {
  digitalWrite(LEFT_A1, HIGH);
  digitalWrite(LEFT_B1, LOW);
  digitalWrite(RIGHT_A2, HIGH);
  digitalWrite(RIGHT_B2, LOW);
}
void backward() {
  digitalWrite(LEFT_A1, LOW);
  digitalWrite(LEFT_B1, HIGH);
  digitalWrite(RIGHT_A2, LOW);
  digitalWrite(RIGHT_B2, HIGH);
}
void left() {
  digitalWrite(LEFT_A1, LOW);
  digitalWrite(LEFT_B1, HIGH);
  digitalWrite(RIGHT_A2, HIGH);
  digitalWrite(RIGHT_B2, LOW);
}
void right() {
  digitalWrite(LEFT_A1, HIGH);
  digitalWrite(LEFT_B1, LOW);
  digitalWrite(RIGHT_A2, LOW);
  digitalWrite(RIGHT_B2, HIGH);
}
void sLeft() {
  digitalWrite(LEFT_A1, LOW);
```

- 어플리케이션 UI, 스플래시 화면, 버튼 제작
- 아두이노에서 4방향 이동 코딩

●김범미

```
1 package com.example.sound;
2
3 import ...
29 public class MainActivity extends AppCompatActivity implements View.OnClickListener, SpeechRecognizerListener{
30
31     private SpeechRecognizerClient client;
32     private static final int REQUEST_CODE_AUDIO_AND_WRITE_EXTERNAL_STORAGE = 0;
33     TextView tv_result;
34     ImageView iv_mic;
35
36     @Override
37     protected void onCreate(Bundle savedInstanceState) {
38         super.onCreate(savedInstanceState);
39         setContentView(R.layout.activity_main);
40
41
42         //주요 권한 체크
43         if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.RECORD_AUDIO) != PackageManager.PERMISSION_GRANTED) {
44             ActivityCompat.requestPermissions( activity: this, new String[] {Manifest.permission.RECORD_AUDIO}, requestCode: 5);
45         }
46
47
48         // SDK 초기화
49         SpeechRecognizerManager.getInstance().initializeLibrary( context: this);
50
51         // 클라이언트 생성
```

-어플리케이션에서 음성 api 인식 기능 구현

●전양희

-어플리케이션에서 블루투스 연결 기능 구현

```
#include <SoftwareSerial.h>

int Tx=6; //전송
int Rx=7; //수신

SoftwareSerial btSerial(Tx, Rx);

void setup()
{
    Serial.begin(9600);
    btSerial.begin(9600);
}

void loop()
{
    if (btSerial.available()) {
        Serial.write(btSerial.read());
    }
    if (Serial.available()) {
        btSerial.write(Serial.read());
    }
}
```

그림 9 아두이노 블루투스 코드

아두이노 보드와 블루투스 모듈 HC-06을 연결해주기 위한 코드이다.

```
compile 'com.akexorcist:bluetoothspp:1.0.0'
```

그림 10 블루투스 라이브러리 추가

안드로이드 스튜디오에서 블루투스를 사용하기 위한 라이브러리를 추가해준다.

```
<uses-permission android:name="android.permission.BLUETOOTH" />
<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
```

그림 11 블루투스 권한 부여

블루투스를 사용하기 위해 블루투스 권한을 부여해준다.

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    bt = new BluetoothSPP( context, this); //initializing

    if (!bt.isBluetoothAvailable()) { //블루투스 사용 불가
        Toast.makeText(getApplicationContext()
            , text: "Bluetooth is not available"
            , Toast.LENGTH_SHORT).show();
        finish();
    }

    bt.setOnDataReceivedListener((data, message) → {
        Toast.makeText( context, MainActivity.this, message, Toast.LENGTH_SHORT).show();
    });

    bt.setBluetoothConnectionListener(new BluetoothSPP.BluetoothConnectionListener() { //연결됐을 때
        public void onDeviceConnected(String name, String address) {
            Toast.makeText(getApplicationContext()
                , text: "Connected to " + name + "\n" + address
                , Toast.LENGTH_SHORT).show();
        }
    })
}
```

그림 12 MainActivity 코드 - 1

```

    public void onDeviceDisconnected() { //연결해제
        Toast.makeText(getApplicationContext()
            , text "Connection lost", Toast.LENGTH_SHORT).show();
    }

    public void onDeviceConnectionFailed() { //연결실패
        Toast.makeText(getApplicationContext()
            , text "Unable to connect", Toast.LENGTH_SHORT).show();
    }
}

Button btnConnect = findViewById(R.id.btnConnect); //연결시도
btnConnect.setOnClickListener((v) → {
    if (bt.getServiceState() == BluetoothState.STATE_CONNECTED) {
        bt.disconnect();
    } else {
        Intent intent = new Intent(getApplicationContext(), DeviceList.class);
        startActivityForResult(intent, BluetoothState.REQUEST_CONNECT_DEVICE);
    }
});

```

그림 13 MainActivity 코드 - 2

```

public void onDestroy() {
    super.onDestroy();
    bt.stopService(); //블루투스 중지
}

public void onStart() {
    super.onStart();
    if (!bt.isBluetoothEnabled()) { //
        Intent intent = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
        startActivityForResult(intent, BluetoothState.REQUEST_ENABLE_BT);
    } else {
        if (!bt.isServiceAvailable()) {
            bt.setupService();
            bt.startService(BluetoothState.DEVICE_OTHER); //DEVICE_ANDROID는 안드로이드 기기 끼리
            setup();
        }
    }
}
}

```

그림 14 MainActivity 코드 - 3


```

public void setup() {
    Button btnSend = findViewById(R.id.btnSend); //데이터 전송
    btnSend.setOnClickListener((v) -> { bt.send( data: "Text", CRLF: true); });
}

public void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == BluetoothState.REQUEST_CONNECT_DEVICE) {
        if (resultCode == Activity.RESULT_OK)
            bt.connect(data);
    } else if (requestCode == BluetoothState.REQUEST_ENABLE_BT) {
        if (resultCode == Activity.RESULT_OK) {
            bt.setupService();
            bt.startService(BluetoothState.DEVICE_OTHER);
            setup();
        } else {
            Toast.makeText(getApplicationContext()
                , text: "Bluetooth was not enabled."
                , Toast.LENGTH_SHORT).show();
            finish();
        }
    }
}
}

```

그림 15 MainActivity 코드 - 4

아두이노 보드와 휴대폰 사이의 블루투스를 동작시켜주기 위해 MainActivity에 코드를 그림 4, 5, 6, 7과 같이 작성해주었다.

○ 향후 진행해야 할 부분

- 아두이노 모듈 정상 작동 테스트
- 어플리케이션 기능 통합
- 아두이노로 펫 명령어 코딩
- 어플리케이션과 펫 연결 및 테스트

4. 개발 일정

[illegible]