

# 아두이노 블루투스 통신

# 아두이노 블루투스 통신

---

## ❖ HC-06

- Bluetooth 2.0
  - iPhone은 bluetooth 3.0 이상 지원



# 아두이노 블루투스 통신

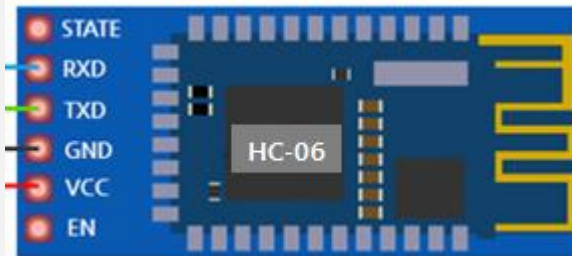
## ❖ 핀연결

RXD     D12

TXD     D13

GND     GND

VCC     5V



# 아두이노 블루투스 통신

---

## ❖ 블루투스 모듈 이름 및 PIN 의 변경

```
#include <SoftwareSerial.h>

SoftwareSerial BTSerial(13, 12); // SoftwareSerial(RX, TX)

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

void loop() {
  if (BTSerial.available()) {
    Serial.write(BTSerial.read()); //블루투스측 내용을 시리얼모니터에 출력
  }
  if (Serial.available()) {
    BTSerial.write(Serial.read()); //시리얼 모니터 내용을 블루투스 측에 WRITE
  }
}
```

## ❖ SoftwareSerial 라이브러리

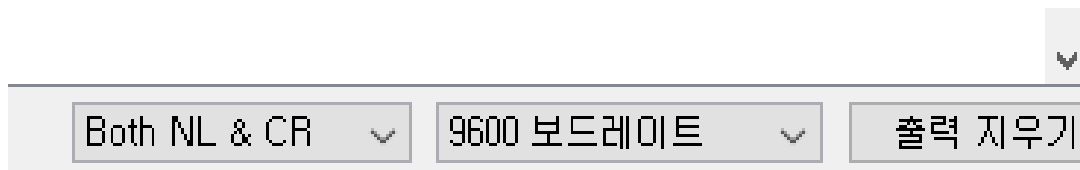
○ <https://www.arduino.cc/en/Reference/SoftwareSerial>

메서드	설명
<b>SoftwareSerial( )</b>	생성자
<b>available( )</b>	수신 데이터 존재 여부
<b>begin( )</b>	전송 속도 설정
<b>listen( )</b>	마스터 모드에서 접속 대기
<b>isListening( )</b>	마스터 모드에서 접속 대기 상태 질의
<b>peek( )</b>	데이터 읽기(제거하지 않음)
<b>read( )</b>	데이터 읽기
<b>print( )</b>	데이터 출력
<b>println( )</b>	데이터 출력
<b>write( )</b>	데이터 출력
<b>overflow( )</b>	버퍼 초과여부

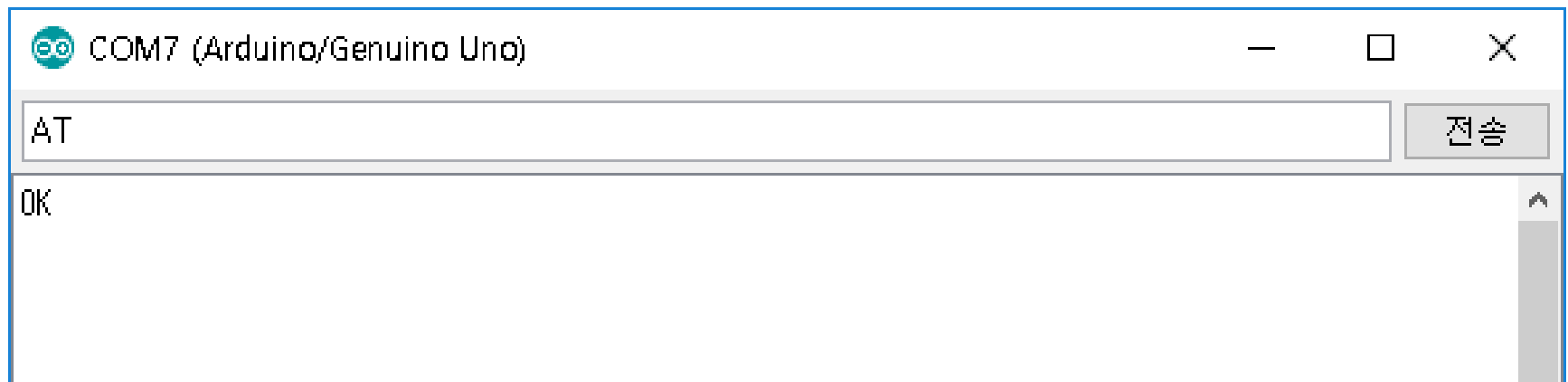
## 아두이노 블루투스 통신

### ❖ 동작확인

- 시리얼 모니터 입력창
  - Both NL&CR 선택



- 'AT' 입력 후 전송
  - 'OK' 답변 확인
- 정상 작동



## ❖ 방법1

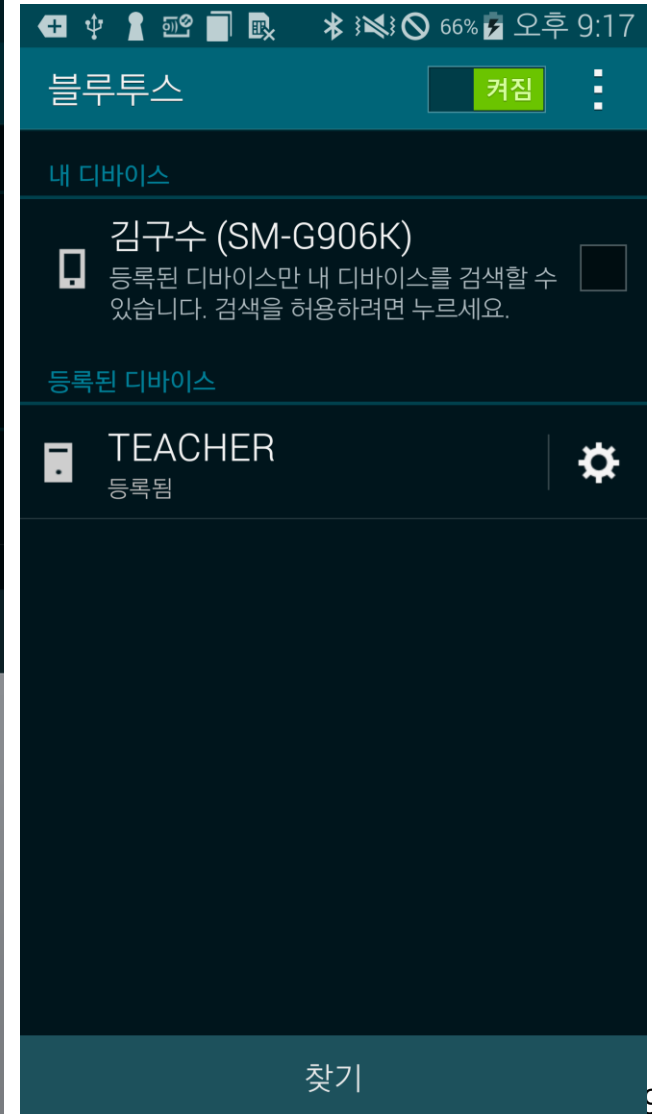
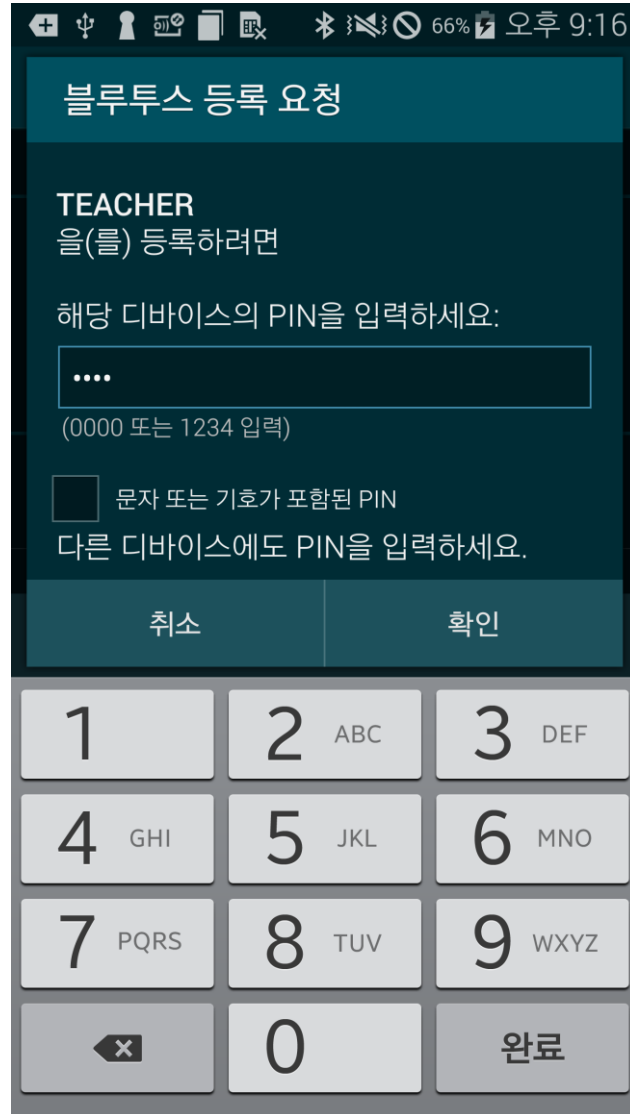
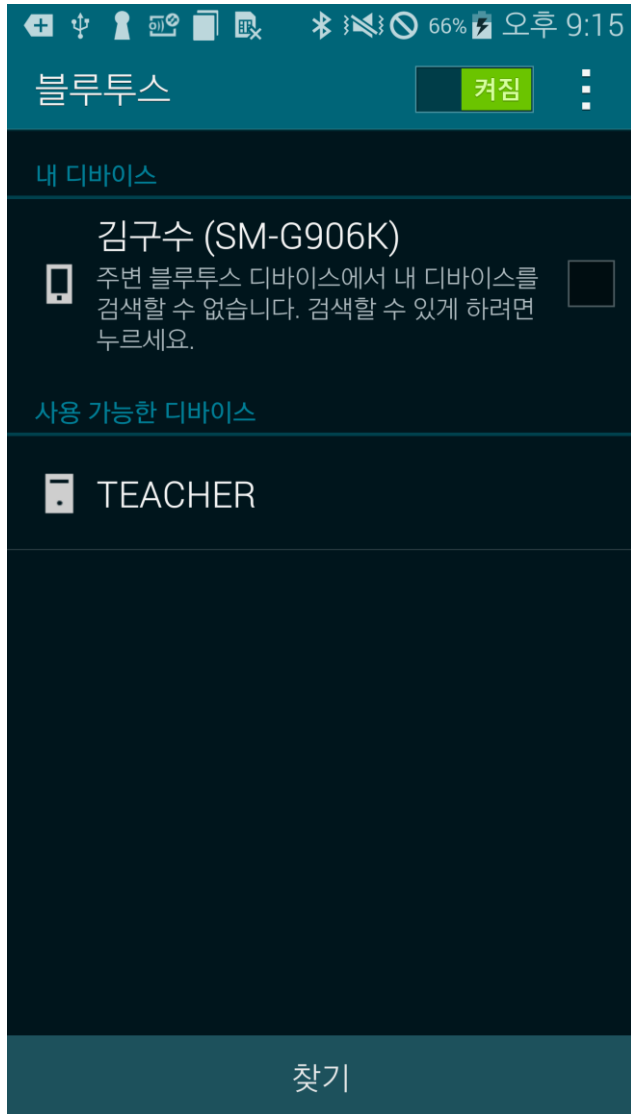
- 이름 확인/변경
  - AT+NAMExxxxx
    - xxxxx라는 이름으로 Bluetooth 모듈 설정
  - AT+NAMEMyHC06  
OKname
- PIN(비밀번호) 설정
  - AT+PINxxxxx
    - xxxx PinNumber설정
  - AT+PIN1234  
OKSetPin

## ❖ 방법2

- 이름 확인/변경
  - AT+NAME?
    - 현재 이름 확인
  - AT+NAME=xxxxx
    - xxxxx라는 이름으로 Bluetooth 모듈 설정
  - AT+NAME=MyHC06  
OK
- PIN(비밀번호) 설정
  - AT+PSWD?
    - 비밀번호 확인
  - AT+PSWD="xxxx"
    - xxxx PinNumber설정
  - AT+PSWD="1234"  
OK



# 아두이노 블루투스 통신



# 아두이노 블루투스 통신

## ❖ Bluetooth Terminal HC-05 앱 설치

Bluetooth Terminal HC-05  
mightyIT  
4.6 ★ (243) • 5만 ↓

간단한 HC-05 / HC-06 터미널 임베디드 시스템에 대한  
데이터를 전송 및 모니터링에

제거 열기

광고 포함 • 인앱 구매

Bluetooth Terminal  
Juan Sebastian Ochoa Zambrano  
4.0 ★ (430) • 10만 ↓

추가 정보 설치

Bluetooth Terminal  
WinGoodHar

Bluetooth Terminal HC-05 SCAN

Paired Devices

TEACHER  
00:21:13:00:E7:E0

여보야, 결혼 재혼을 위한 ...  
이 앱은 결혼가임비 무료

설치

Bluetooth Terminal HC-05  
Connected to TEACHER

Click here to change Received data  
format: ASCII or HEX.

GOT IT

Auto Scroll

Enter ASCII Command Send ASCII

Btn 1 Btn 2 Btn 3 Btn 4 Btn 5

데일리/ 위클리 할인 정보 \$40 이상  
주문 무료 배송

설치

## 아두이노 블루투스 통신

---

### ❖ ex01/app.ino 스마트폰과 블루투스 모듈과의 통신을 통한 문자열 전송

```
#include <SoftwareSerial.h>
#include <MiniCom.h>

MiniCom com;

SoftwareSerial BTSerial(13, 12); // SoftwareSerial(RX, TX)

String myString="";    //받는 문자열
int count=0;

void setup() {
    com.init();
    com.print(0, "[[Bluetooth]]");
    BTSerial.begin(9600); //블루투스 시리얼 개방
}
```

### ❖ ex01/app.ino 스마트폰과 블루투스 모듈과의 통신을 통한 문자열 전송

```
void loop() {  
    while(BTSerial.available()) { //BTSerial에 전송된 값이 있으면  
        //BTSerial int 값을 char 형식으로 변환  
        char myChar = (char)BTSerial.read();  
        //수신되는 문자를 myString에 모두 붙임 (1바이트씩 전송되는 것을 연결)  
        myString+=myChar;  
        delay(5); //수신 문자열 끊김 방지  
    }  
  
    if(!myString.equals("")) { //myString 값이 있다면  
        myString.trim();  
        char buf[17];  
        sprintf(buf, "%d)%s", ++count, myString.c_str());  
        com.print(1, buf);  
        BTSerial.println("input value: "+myString);  
        myString=""; //myString 변수값 초기화  
    }  
}
```

# BtMiniCom 클래스

---

## ❖ BtMiniCom

- 블루투스 통신 지원 클래스
- MiniCom 클래스 상속

# BtMiniCom 클래스

## ❖ ex02/BtMiniCom.h

```
#pragma once

#include <Arduino.h>
#include <SoftwareSerial.h>
#include <MiniCom.h>

typedef void (*btminicom_callback_t)(String message);

class BtMiniCom : public MiniCom {
protected:
    SoftwareSerial btSerial;
    btminicom_callback_t callback; // 메시지 수신시 호출할 콜백 함수 포인터

public:
    BtMiniCom(int rx, int tx, btminicom_callback_t callback=NULL,
               long serial_bps=115200, int lcd_addr=0x27);
    void init();
    String readLine();           // \r\n를 제외한 문자열을 리턴
    void send(String msg);       // 메시지 전송
    void run();
};
```

# BtMiniCom 클래스

---

## ❖ ex02/BtMiniCom.cpp

```
#include "BtMiniCom.h"

BtMiniCom::BtMiniCom(int rx, int tx, btminicom_callback_t callback,
                    long serial_bps, int lcd_addr)
    : MiniCom(serial_bps, lcd_addr), btSerial(rx, tx), callback(callback) {
}

void BtMiniCom::init() {
    MiniCom::init();
    btSerial.begin(9600);
}

String BtMiniCom::readLine(){ // \r\n를 제외한 문자열을 리턴
    String message="";
    while(btSerial.available()) {
        char data = (char)btSerial.read();
        message+=data;
        delay(5);    //수신 끊김 방지 ***
    }
    message.trim();
    return message;
}
```

## BtMiniCom 클래스

---

### ❖ ex02/BtMiniCom.cpp

```
void BtMiniCom::send(String msg){
    btSerial.println(msg);
}

void BtMiniCom::run(){
    String msg = readLine();
    if(msg != "" && callback != NULL) {
        callback(msg);
    }
    MiniCom::run();
}
```



# BtMiniCom 클래스

---

## ❖ ex02/app.ino

```
#include <BtMiniCom.h>

void receive_msg(String msg);

BtMiniCom com(13, 12, receive_msg);
int count=0;

void receive_msg(String msg) {
    char buf[17];
    sprintf(buf, "%d)%s", ++count, msg.c_str());
    com.print(1, buf);
    com.send("input value: " + msg);
}

void setup() {
    com.init();
    com.print(0, "[[Bluetooth]]");
}

void loop() {
    com.run();
}
```

---

❖ 서보모터 3번 연결

❖ 블루투스 통신으로 서보 모터 제어

❖ APP

- Btn 3 --> Open / "open"      --> 서보 모터 90도 회전
- Btn 4 --> Close / "close"      --> 서보 모터 0도 회전

---

## ❖ App으로 도어락 열기

- "doorlock xxxxx" 전송
- EEPROM에 저장된 비밀번호를 검사해서
  - 일치하면 서보모터 90도 회전
  - 5초 후에 서보모터 0도 회전

## ❖ Btn 5

- Door : "doorlock xxxx"