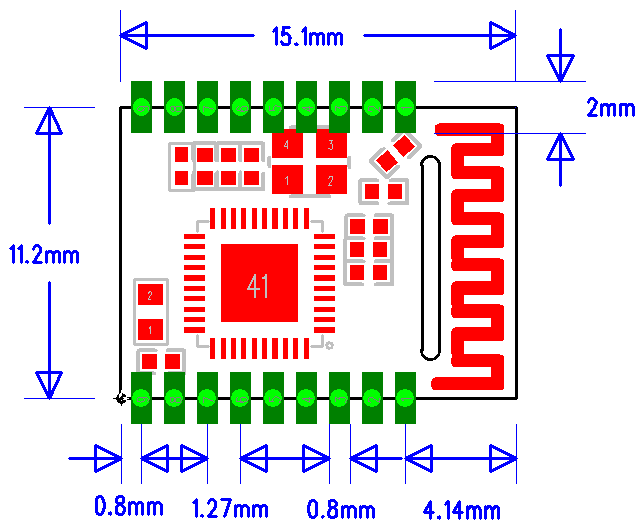
**RSBRS02AA(BLE 4.2) Module UART 통신**

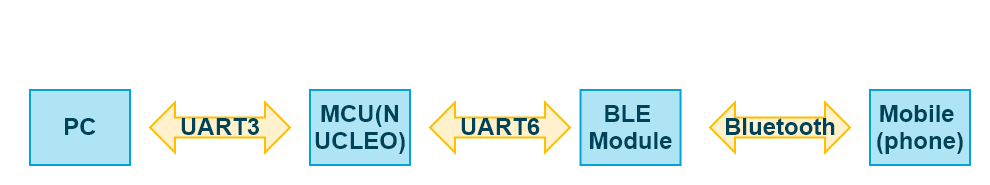
**프로젝트 구성 요소**

* BLE모듈: RSBRS02AA(BLE 4.2)
* 개발 보드: NUCLEO-F429ZI(STM32F429ZIT6U)
* 통신 방식: UART
* SSH 클라이언트(통신 연결 프로그램) : Xshell

**BLE 모듈**



*Module PIN* RSBRS02AA



<통신 연결도>

**동작 흐름** ※ 3페이지 flow chart 참고.

* CUBE MX로 NUCLEO보드 클럭 등 초기 셋팅.
* 모듈에 블루투스 기기(폰) 연결되면 “TTM:CONNECT” 출력하고 수신 인터럽트(Receive IT)대기
* 모듈이나 PC에서 데이터 들어오면 ‘stm32f4xx\_it.c’에 IRQHandler를 거쳐서 RxCallback함수로 넘어 감.
* 입력된 데이터가 UART6(Module), UART3(PC) 중 어느 것에서 왔는지 확인.
* 해당되는 UART의 명령을 수행. (ex. Uart3-> BRTS ON(Low), BCTS OFF(HIGH), uart6(BLE)로 Transmit (데이터 전송) )
* 전송 끝나면 uart3, uart6 인터럽트 대기 시키고 BRTS off, BCTS on (But, uart3끝나고 BRTS off하면 작동이 제대로 안되는데 이유를 못찾겠음.)

**추가 설명**

* HAL드라이버를 쓰면 UART Receive Interrupt시 받는 데이터의 길이를 정해놔야 함. 길이보다 짧으면 인터럽트 발생이 안되고, 길면 정해진 문자열 길이만큼만 받음. (링버퍼로 해결할 수 있을 것 같은데 아직 제대로 적용 못해 봄)
* 그래서 원하는 데이터를 넘기고 받기 위해 우리만의 프로토콜을 만듬. 예를 들어 BLE에서 ‘0’을 입력하면 “BLE:I’m0” PC 쉘 창에 출력, PC에서 ‘1’을 작성하면 핸드폰에 “PC:I’m1”출력됨.
* Uart3랑 Uart6에서 데이터를 다르게 처리 하는 이유는 다양한 방법이 있다는 것을 보여주기 위함. 두 방법 중 아무거나 써도 무관
* 아래 첨부한 PIN Definition 테이블을 보면 알 수 있듯이, BRTS를 켜 놓으면(Low상태) 항상 Rx mode이고 전력소모가 큼.
* 그래서 uart3 끝나고 BRTS off 시켜도 작동 될 수 있게 하는 방법 찾아야함.
* Pin 정의 테이블상에 나와있는 여분 I/O Pin은 사용 불가

**개선 필요 사항**

* BRTS off 상태에서도 연속해서 작동되도록 개선 필요.
* 가변적 길이의 데이터 받아서 처리할 수 있도록 개선 필요.

**Flow Chart**

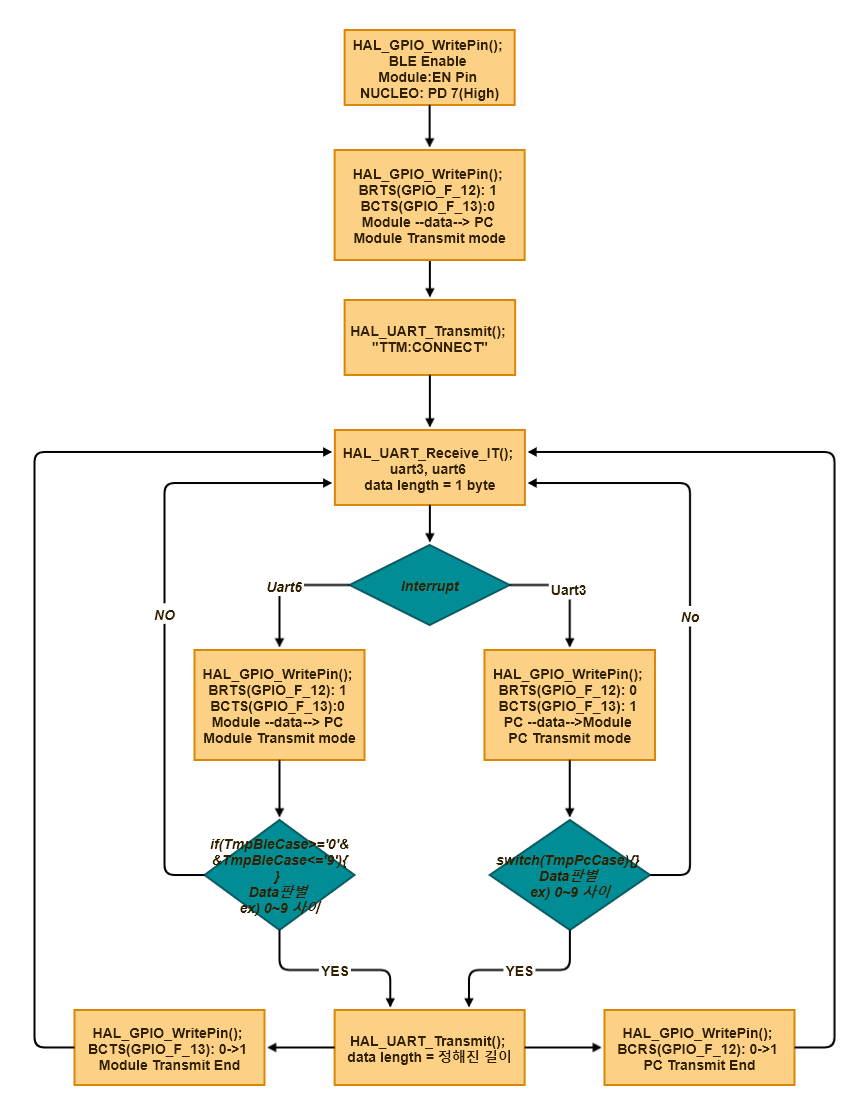


Figure <BLE - Uart 통신 Flow Chart>

**RSBRS02AA Pin Definition**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pin No. | Module Pin Name | Chip Pin Name | I/O | Specification |
| Pin1 | GND | GND | － | Grounding GND |
| Pin2 | VCC | VCC | － | Power Supply: 1.6~3.6V |
| Pin3 | IO7 | P15 | O | Sleep Mode indicator |
| Pin4 | IO6 | P24 | O | Connection Status indicator |
| Pin5 | RES | RES | I | * Reset Input - Active when low level, with internalpull-up. |
| Pin6 | EN | P06 | I/O | Module-enabled control line *(level trigger mode as default)*   * Level trigger mode - Active when low level, with internalpull-up.   0: Module starting to broadcast, until connected to the mobile device  1: Entering sleep mode immediately, regardless of the current status |
| Pin7 | SWC | SWC | － | Clock Pin when download firmware |
| Pin8 | SWD | SWD | － | Data Pin when download firmware |
| Pin9 | I2C- | P21 | I/O |  |
| Pin10 | IO3 | P10 | I/O |  |
| Pin11 | IO2 | P11 | I/O |  |
| Pin12 | IO1 | P12 | I/O |  |
| Pin13 | IO0 | P14 | I/O |  |
| Pin14 | BRTS | P16 | I | As the data sending requests (for module wake-up)  0: Host has data to send, and module will wait for data transmission from the host so will not sleep  1. Host has no data to send, or data has been sent. So the value of the signal should be set at “1”. |
| Pin15 | BCTS | P17 | O | Data input signal (for host wake-up, optional)  0: Module has data to send, and the host will receive the data.  1: Module has no data to send, or data has been sent, and the value of the signal will be set at “1”. |
| Pin16 | TX | P27 | O | Serial port TX |
| Pin17 | RX | P28 | I | Serial port RX |
| Pin18 | ADC | P03 | I | Analog acquisition |