
Chapter 129. 컴퓨터와의 시리얼 연결

ATmega328에서 사용할 수 있는 시리얼 통신의 종류에는 UART, SPI, I2C 등 여러 가지가 있다. 이 중 UART는 가장 오랜 역사를 가지고 있으며 다양한 장치에서 채택하고 있어 1:1 통신만이 가능하다는 한계에도 불구하고 많이 사용된다. UART는 마이크로컨트롤러 사이의 통신, 마이크로컨트롤러와 주변장치 사이의 통신뿐만 아니라 마이크로컨트롤러와 컴퓨터와의 통신에서도 흔히 사용된다. 하지만 컴퓨터는 UART 통신을 지원하지 않으며 D-SUB 단자를 통한 RS232나 USB를 통한 시리얼 통신을 사용하여야 하므로 UART와 RS232/USB 사이의 변환을 위한 별도의 변환 모듈이 필요하다. 먼저 ATmega328의 UART 통신부터 살펴보자.

1. ATmega328의 UART 통신

아두이노에서 흔히 사용하는 시리얼 통신은 UART 통신이라고 앞서서도 설명하였다. UART 통신은 RX(receive), TX(transmit), GND(ground) 3개의 연결선만으로 통신이 가능하다. UART는 3.3V 또는 5V 로직을 주로 사용하며 ATmega328은 5V 로직을 사용한다. 5V 로직을 사용하는 다른 주변장치는 ATmega328과 직접 연결하여 사용이 가능하지만 3.3V 로직을 사용하는 주변장치와의 연결에서는 주변장치에서 입력으로 받아들일 수 있는 전압 범위를 반드시 확인하여야 한다. 3.3V와 5V 로직이 흔히 사용되므로 두 가지 전압 범위를 모두 허용하는 장치가 많이 존재하지만, 일부 3.3V 장치의 경우 5V 전압이 가해지는 경우 파손될 수 있으므로 주의하여야 한다. 이 경우 그림 1에서와 같이 간단한 전압 분배 (voltage divider) 회로를 통해 전압을 3.3V로 맞추으로써 두 장치를 연결할 수 있다.

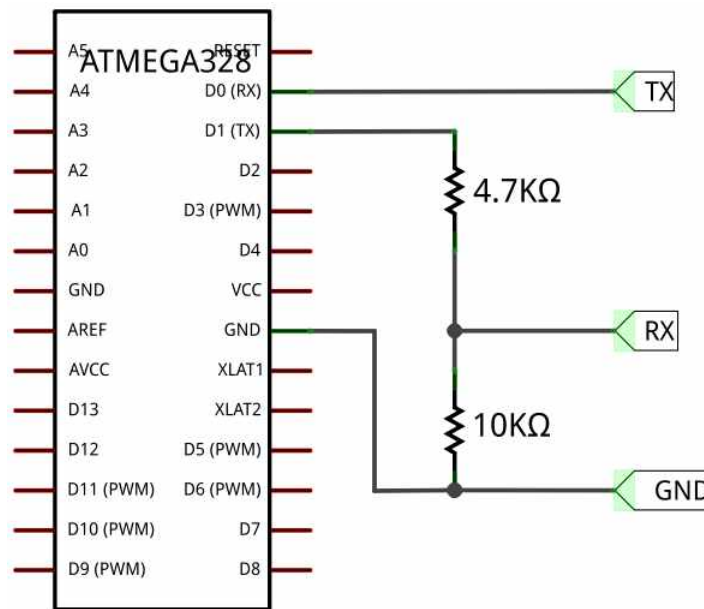


그림 1. 3.3V 로직을 사용하는 주변장치와 UART 연결

반면 ATmega328의 RX의 경우 기준 전압인 5V의 60%인 3V에서도 HIGH로 인식하므로 3.3V 로직을 사용하는 주변장치의 TX를 직접 연결하여도 크게 문제는 없다. 하지만 잡음이나 전압 감쇄에 의해 주변장치로부터의 전압이 3V 이상임이 보장되지 못한다면 3.3V와 5V 사이의 논리 레벨 변환기(logic level shifter)를 사용하는 것이 안전하다.¹⁾

ATmega328에 UART 시리얼 통신 장치를 연결하기 위해서는 PD0 핀(디지털 0번 핀)과 PD1 핀(디지털 1번 핀)을 사용하며 이를 아두이노에서는 하드웨어 시리얼이라고 부른다. 하드웨어 시리얼은 UART 통신 기능을 하드웨어로 구현한 것에 해당하며 이와 대비되는 개념으로 소프트웨어 시리얼 역시 존재한다. 소프트웨어 시리얼은 UART 통신을 소프트웨어를 통해 구현해 놓은 것으로 아두이노 우노의 경우 소프트웨어 시리얼 클래스인 SoftwareSerial을 통해 모든 핀을 UART 통신용으로 사용하는 것이 가능하다. 아두이노 우노의 경우 20개의 입출력 핀이 존재하므로 최대 10개의 UART 통신장치를 연결하는 것이 가능하다. 하드웨어 시리얼 클래스가 Serial인 점을 제외하면 소프트웨어 시리얼과 기능적인 면에서 차이는 없으므로 여기서는 하드웨어 시리얼 통신만을 다룬다.

아두이노 우노의 경우 ATmega328의 UART 시리얼 통신을 위한 포트에는 또 다른 마이크로 컨트롤러가 연결되어 있다. 컴퓨터의 USB 연결을 통해 전달되는 데이터를 UART 통신을 위한 데이터로 변환해주기 위한 ATmega16u2 마이크로컨트롤러가 바로 그것이다. 아두이노 우노 보드를 USB 케이블로 컴퓨터에 연결하면 장치관리자의 포트 부분에 직렬 포트(COM)가 생

1) 아두이노 우노 보드는 5V를 기준으로 하므로 이 책에서는 5V 기준 전압을 위주로 설명한다.

성되며 이 직렬 포트는 ATmega16u2 마이크로컨트롤러에 해당한다. 컴퓨터에 아두이노 우노 보드를 연결하였을 때 나타나는 직렬 포트는 시리얼 방식으로 프로그램을 업로드를 위해 사용될 뿐만이 아니라, 컴퓨터와 아두이노 우노 사이의 시리얼 통신을 위해서도 사용된다. 컴퓨터와 아두이노 우노 사이의 시리얼 통신은 시리얼 모니터를 통해 송수신되는 데이터를 확인할 수 있다.

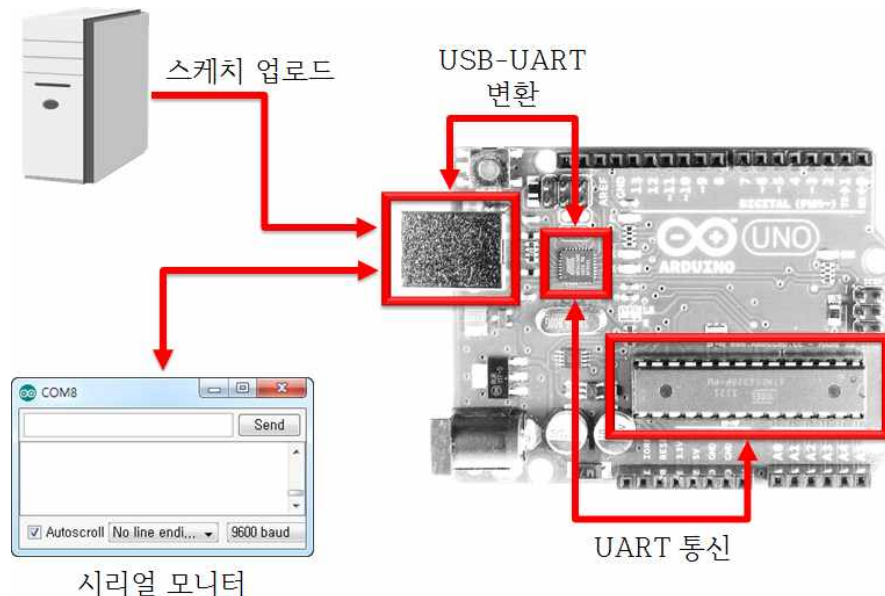


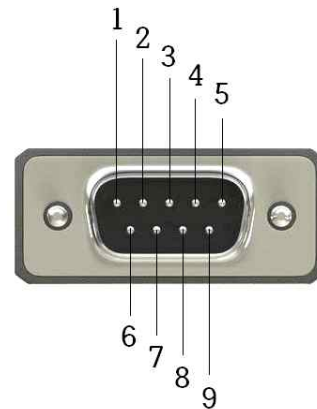
그림 2. 아두이노 우노와 컴퓨터 사이의 USB 연결

2. RS232-UART 변환기

컴퓨터에서의 시리얼 통신은 COM 포트를 통해 이루어지며 RS232 프로토콜을 따른다. RS232는 Recommended Standard System for Serial Communication의 일부분으로 1969년 미국 전자공학협회에서 제정한 통신 인터페이스 규약의 하나이다. RS232는 20K 보레이트 이하의 속도로 15m 내에서 직렬로 데이터를 전송할 수 있다. RS232 연결을 위해서는 9핀의 DB9 인터페이스가 많이 사용되며 0은 +3V 이상, 1은 -3V이하로 반전된 논리를 사용한다. 그림 3은 DB9 커넥터의 핀 배치를 나타낸 것이며 각 핀에 대한 설명은 표 1에 나타나 있다.



(a) DB9 커넥터



(b) DB9 커넥터 핀 배치

그림 3. DB9 커넥터

핀 번호	이름	약어	기능
1	DCD	Data Carrier Detect	모뎀과 같은 데이터 통신 장치(data communication equipment, DCE)가 다른 DCE와 접속이 완료되었을 때 상대방 DCE는 캐리어 신호를 보내며 이 신호를 검출하였을 때 컴퓨터와 같은 데이터 단말 장치(data terminal equipment, DTE)에 알려주는 신호선
2	RXD	Receive Data	외부 장치로부터 데이터를 입력받는 신호선
3	TXD	Transmit Data	외부로 데이터를 출력하는 신호선
4	DTR	Data Terminal Ready	자신이 송수신이 가능한 상태임을 알리는 신호선. 전원 인가 후 통신 포트를 초기화한 후 출력한다.
5	GND	Ground	접지
6	DSR	Data Set Ready	자신이 송수신이 가능한 상태임을 알리는 신호선. 전원 인가 후 통신 포트를 초기화하고 이상이 없는 경우 출력한다.
7	RTS	Ready To Send	DTE가 DCE에게 데이터를 받을 준비가 되었음을 알리는 신호선
8	CTS	Clear To Send	DCE 장치가 DTE 장치로 데이터를 받을 준비가 되었음을 알리는 신호선
9	RI	Ring Indicator	상대편 장치가 통신을 위해 전화를 걸어올 때 이 신호를 모뎀이 인식하여 DTE에 알려주는 신호선

표 1. DB9 커넥터 핀

UART의 통신에서는 RS232C 프로토콜에 정의된 핀 중에서 RXD (또는 RX), TXD (또는 TX), GND 세 개의 핀만을 사용한다. RS232 프로토콜은 반전된 로직을 사용하므로 1이 +5V, 0이 0V를 나타내는 ATmega328의 USART와 직접 연결은 불가능하다. 따라서 별도의 장치를 통해 전압 레벨을 맞추어 주어야 하며 이를 라인 드라이버 또는 논리 레벨 변환기라고 부른다. RS232와 UART 사이의 논리 레벨 변환을 위해서는 MAX3232 칩이 흔히 사용되며 MAX3232 칩을 사용한 변환 모듈이 많이 판매되고 있다.

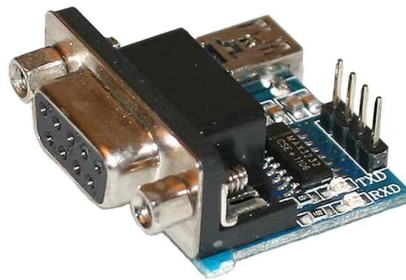


그림 4. MAX3232 칩을 사용한 RS232-UART 논리 레벨 변환 모듈

RS232-UART 변환 모듈을 그림 5와 같이 ATmega328에 연결해 보자. 이 때 주의할 점은 변환 모듈의 RX가 ATmega328의 RX와 연결되어 있다는 점이다. 일반적으로 두 장치를 UART 통신으로 연결하는 경우에는 RX와 TX를 서로 교차하여 연결한다. 하지만 그림 4에 나타난 변환 모듈과 같은 일부 장치는 주변 장치와 RX-RX, TX-TX로 직접 연결하도록 ‘표시만’ 바꾸어 놓은 경우가 있으므로 연결 시에 주의하여야 한다. DB9 커넥터는 컴퓨터로 연결한다.

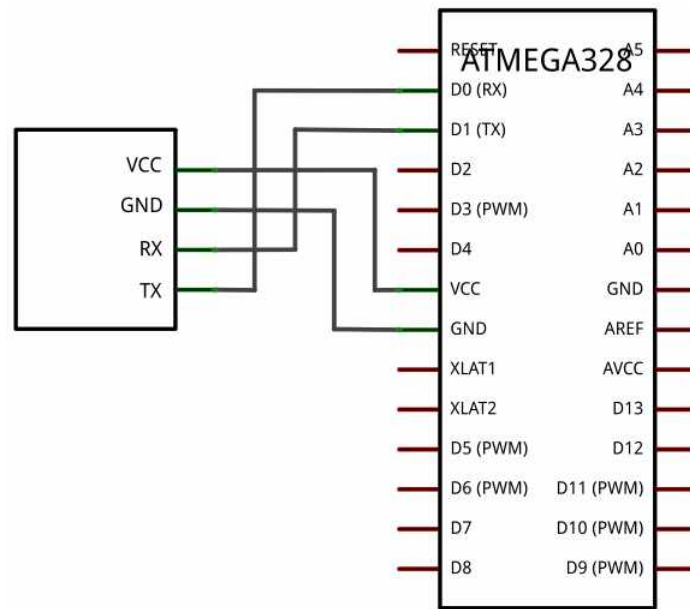


그림 5. RS232-UART 변환 모듈 연결 회로도

컴퓨터의 시리얼포트는 일반적으로 COM1과 COM2 2개가 제공되는 경우가 대부분이지만 컴퓨터에 따라 제공하는 수가 다를 수 있으므로 장치관리자에서 확인하도록 한다. 코드 1은 RS232 - 시리얼 변환기를 통해 '0'에서 '9'까지 문자 데이터를 1초 간격으로 반복적으로 컴퓨터로 전송하는 코드의 예이다.

코드 1. RS232 - 시리얼 변환

```
#define F_CPU 16000000L
#include <avr/io.h>
#include <util/delay.h>
#include "UART.h"

int main(int argc, char *argv[])
{
    char data = '0';

    UART_INIT();                // UART 통신 초기화
    while (1) {
        UART_transmit(data++);  // 카운터값을 컴퓨터로 전달
    }
}
```

```
    UART_transmit('\n');           // 줄바꿈

    if(data > '9') data = '0';

    _delay_ms(1000);
}

return 0;
}
```

코드에서 알 수 있듯이 UART 시리얼 통신을 사용하는 경우와 차이가 없다. 코드 2는 코드 1을 아두이노 스타일로 작성한 코드의 예이다.

코드 2. RS232 - 시리얼 변환 - 아두이노

```
char data = '0';

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

void loop(){
    Serial.println(data++);
    if(data > '9') data = '0';

    delay(1000);
}
```

코드 1과 코드 2의 실행 결과는 그림 6으로 동일하다.

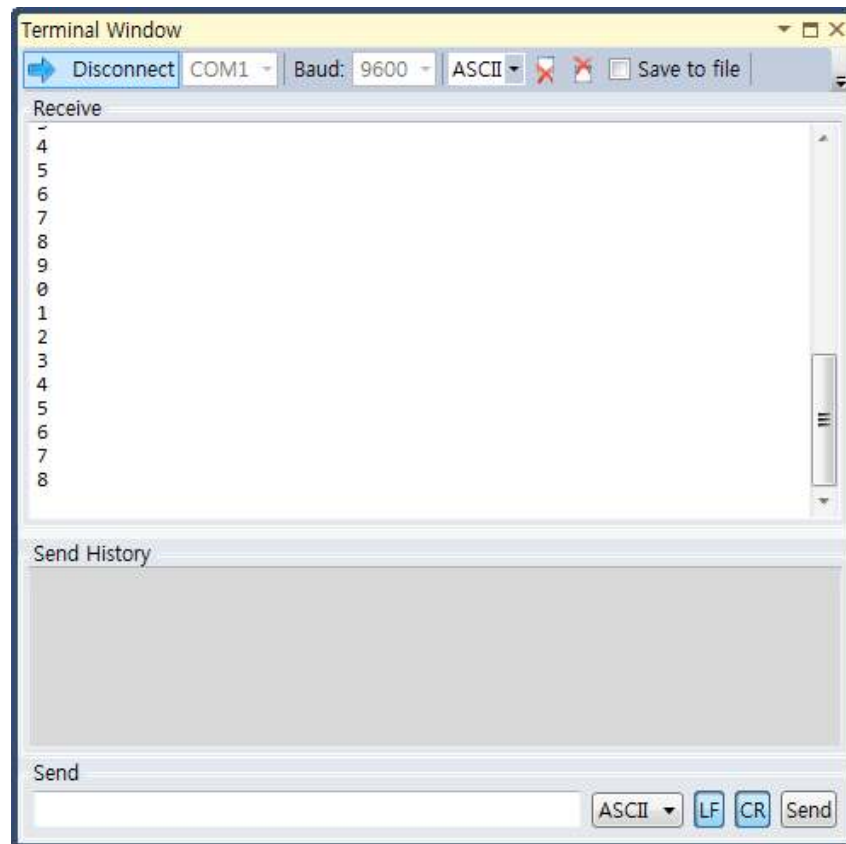


그림 6. 코드 1 및 코드 2 실행 결과 - RS232-UART 변환

무엇이 달라진 것일까? 바로 통신 포트의 번호이다. UART 시리얼 통신을 설명할 때 사용한 포트 번호는 USB-UART 변환을 위한 전용 마이크로컨트롤러에 할당된 포트 번호를 사용하였다. 하지만 코드 1과 코드 2에서는 컴퓨터의 하드웨어 시리얼 포트인 1번 포트를 사용하고 있다. 즉, 마이크로컨트롤러에 업로드 하는 프로그램에는 차이가 없으며 데이터를 수신하는 컴퓨터의 포트만이 달라졌을 뿐이다.

3. USB-UART 변환기

지금까지 컴퓨터와 시리얼 통신을 수행하는 두 가지 방법을 살펴보았다. USB 연결을 UART 연결로 바꾸는 것이 그 하나이며, RS232 연결을 UART 연결로 바꾸는 것이 다른 하나이다. USB-UART 변환을 위해서는 아두이노 우노에 포함되어 있는 ATmega16u2 마이크로컨트롤러가 사용되었고, RS232-UART 변환을 위해서는 MAX3232 칩이 사용되는 등 두 가지 경우 모두에서 신호의 상호 변환을 위해서 전용 하드웨어가 필요하였으며, 이는 이 절에서 설명할 또 다른 USB-UART 변환 장치 역시 마찬가지다.

이미 USB-UART 변환 장치를 살펴보았지만 이는 아두이노 우노에 포함되어 있는 마이크로컨트롤러로 다른 곳에서는 흔히 볼 수 없는 하드웨어이다. 따라서 ATmega328로 기본 회로를 구성하는 경우에는 동일한 기능을 하는 다른 하드웨어 장치를 사용하는 것이 일반적이며 FTDI 칩을 사용한 변환 장치가 흔히 사용된다. USB-UART 변환 장치는 4핀에서 6핀까지 다양한 연결 커넥터를 가지고 있는 제품이 판매되고 있으며 RS232-UART 변환 모듈과 마찬가지로 4개의 핀(RX, TX, VCC, GND)만 있으면 컴퓨터와 시리얼 통신을 수행할 수 있다. 5핀 이상의 커넥터를 가지는 USB-UART 변환 장치의 다른 한 핀은 RS232에서의 DTR 신호 출력을 위해 사용되는 것이 일반적이며 이는 아두이노에서 프로그램의 자동 업로드를 위해 사용된다. 실험에 사용한 USB-UART 변환 모듈은 6개의 연결 핀을 가지고 있으며 이 중 하나는 사용되지 않는 핀이다.



그림 7. FTDI 칩을 사용한 USB-UART 논리 레벨 변환 모듈

ATmega328과의 연결 방법은 그림 5와 기본적으로 동일하며 그림 7의 변환 모듈의 경우 RX-TX로 교차 연결하도록 표시되어 있다. 사용할 코드는 코드 1 및 코드 2를 그대로 사용하면 되며 실행 결과는 그림 8과 같다.

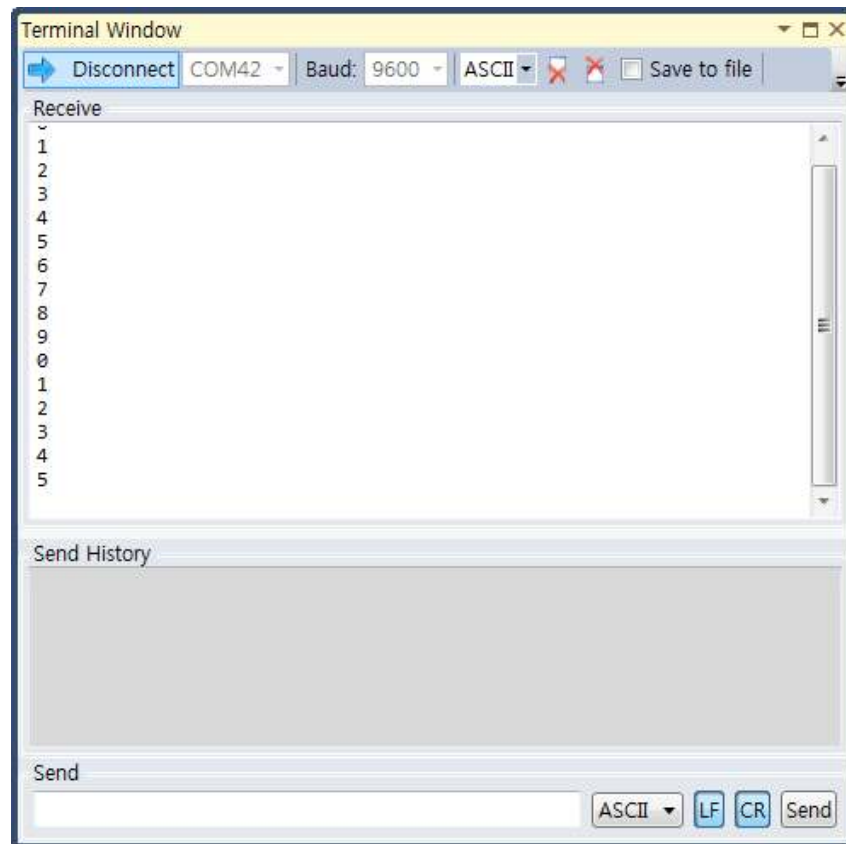


그림 8. 코드 1 및 코드 2 실행 결과 - USB-UART 변환

그림 6과 그림 8은 동일한 코드를 사용하여 얻은 두 가지 결과이다. 무엇이 다른지 발견하였는가? 두 경우에서 컴퓨터와 연결된 시리얼 포트 번호가 RS-232-UART 변환 장치의 경우 COM1, USB-UART 변환 장치의 경우 COM42로 다른 점을 제외하면 동일하다. 마찬가지로 아두이노 우노 보드의 ATmega16u2 마이크로컨트롤러를 사용하는 경우 컴퓨터와 연결된 시리얼 포트 번호가 COM35번 인 점을 제외하면 위의 두 예에서와 동일하다.

4. 컴퓨터와의 시리얼 연결

컴퓨터와 ATmega328을 시리얼로 연결하는 세 가지 방법을 살펴보았다. 세 가지 방법은 모두 전용의 하드웨어 장치를 사용하며, 컴퓨터 쪽에서는 직렬 통신 포트(COM)를 통해 이루어진다는 점에서 동일하다. 하지만 각각 서로 다른 하드웨어를 사용하므로 직렬 통신 포트의 번호는 모두 다르다. 또한 RS232-UART 변환 장치의 경우 컴퓨터의 하드웨어 시리얼 포트를 사용하지만, USB-UART 변환 장치의 경우 컴퓨터에서 생성되는 가상의 시리얼 포트를 사용한다는 점에서 차이가 있다.

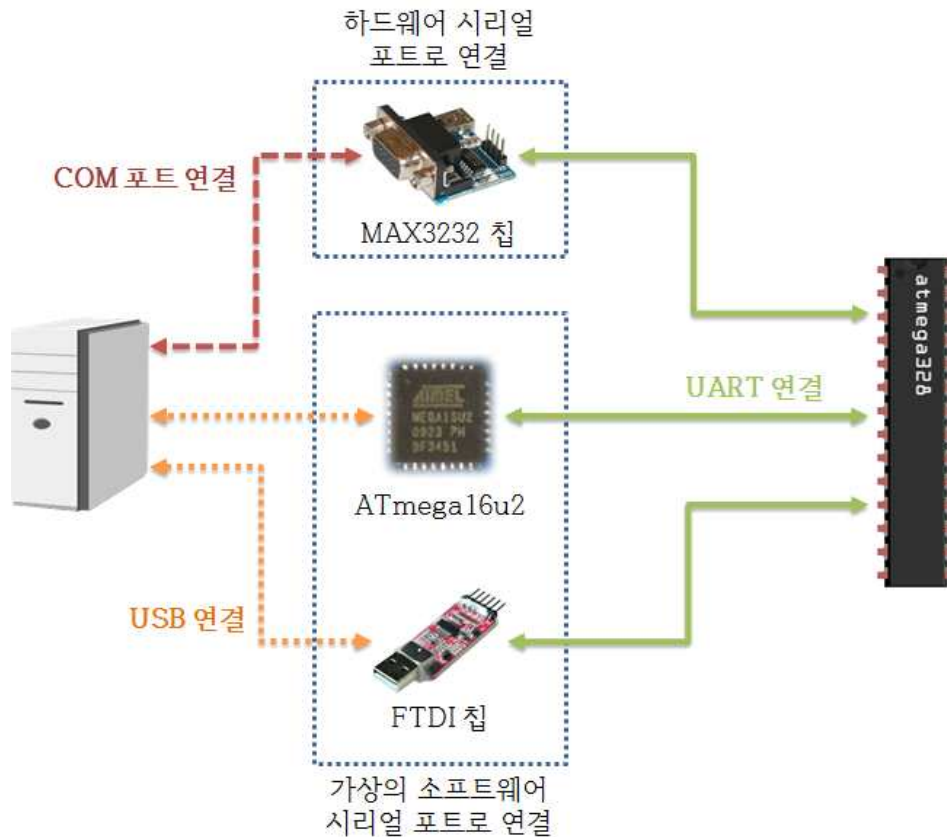


그림 9. 컴퓨터와 ATmega328의 시리얼 연결

실험에 사용한 컴퓨터의 직렬 통신 포트들을 장치관리자에서 확인하는 것으로 이 장을 끝내려고 한다. 장치관리자에는 모두 5개의 직렬 포트가 나타나고 있다. 이 중 COM1과 COM2는 하드웨어 시리얼 포트 RS232-UART 변환 장치를 통해 ATmega328과 연결될 수 있다. COM35는 아두이노 우노에 포함되어 있는 ATmega16u2 마이크로컨트롤러에 할당된 가상의 시리얼 포트이며, COM42는 FTDI 칩에 할당된 가상의 시리얼 포트이다. 마지막 COM39는 ISP 방식의 프로그램 업로드 장치에 할당된 가상의 시리얼 포트이므로 이 포트를 통해서만 컴퓨터와 시리얼 통신을 수행할 수 없다.

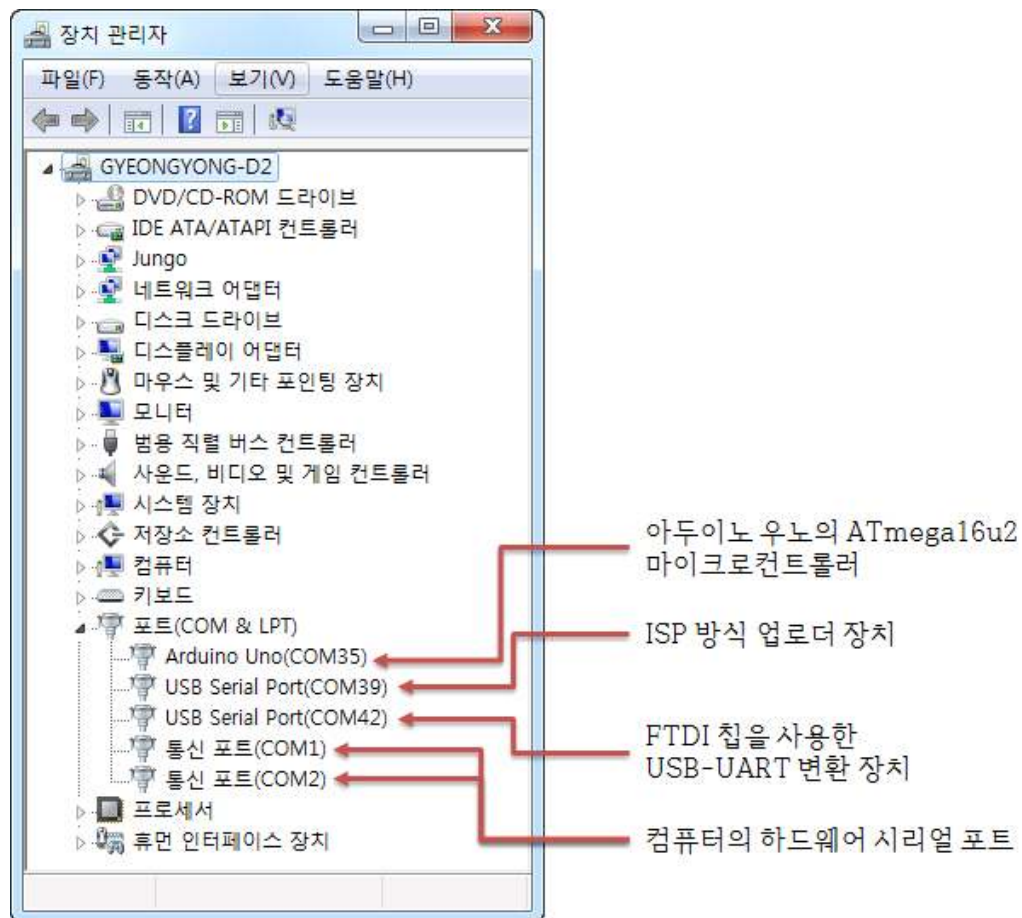


그림 10. 장치관리자의 시리얼 포트 목록