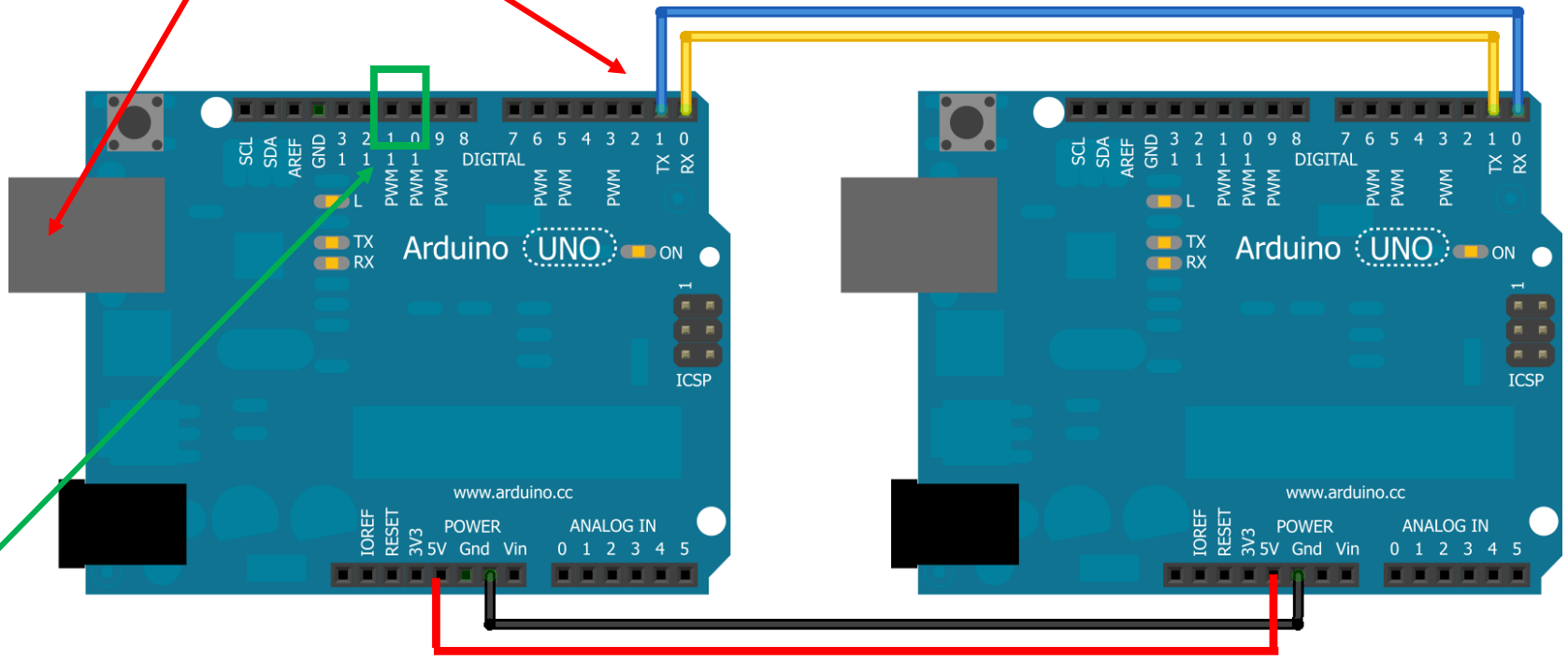
Software Serial Setup

Arduino to Arduino Communication

Interference on
Pin 0 (RX), Pin 1 (TX)

Computer Uses
Hardware Serial
pin 0 (RX), pin 1 (TX)

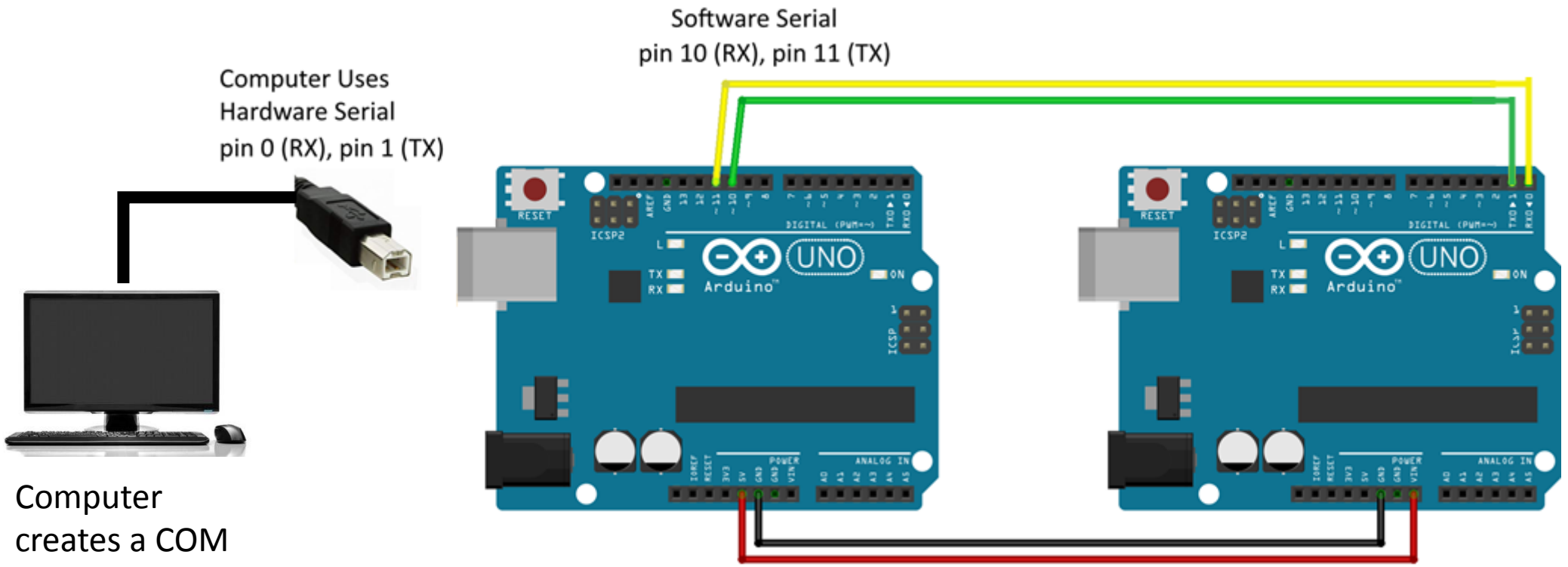


Use Software Serial!

Arduino Sample Code


```
#include <SoftwareSerial.h>

SoftwareSerial BTSerial(10, 11); // RX | TX

void setup()
{
  Serial.begin(9600);
  Serial.println("Software Serial Mode");
  BTSerial.begin(9600);
}

void loop()
{

  // Keep reading from HC-05 and send to Arduino Serial Monitor
  if (BTSerial.available())
    Serial.write(BTSerial.read());

  // Keep reading from Arduino Serial Monitor and send to HC-05
  if (Serial.available())
    BTSerial.write(Serial.read());
}
```

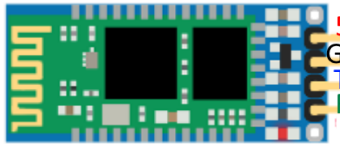
Software Serial Wiring



Computer creates a COM port for USB communication

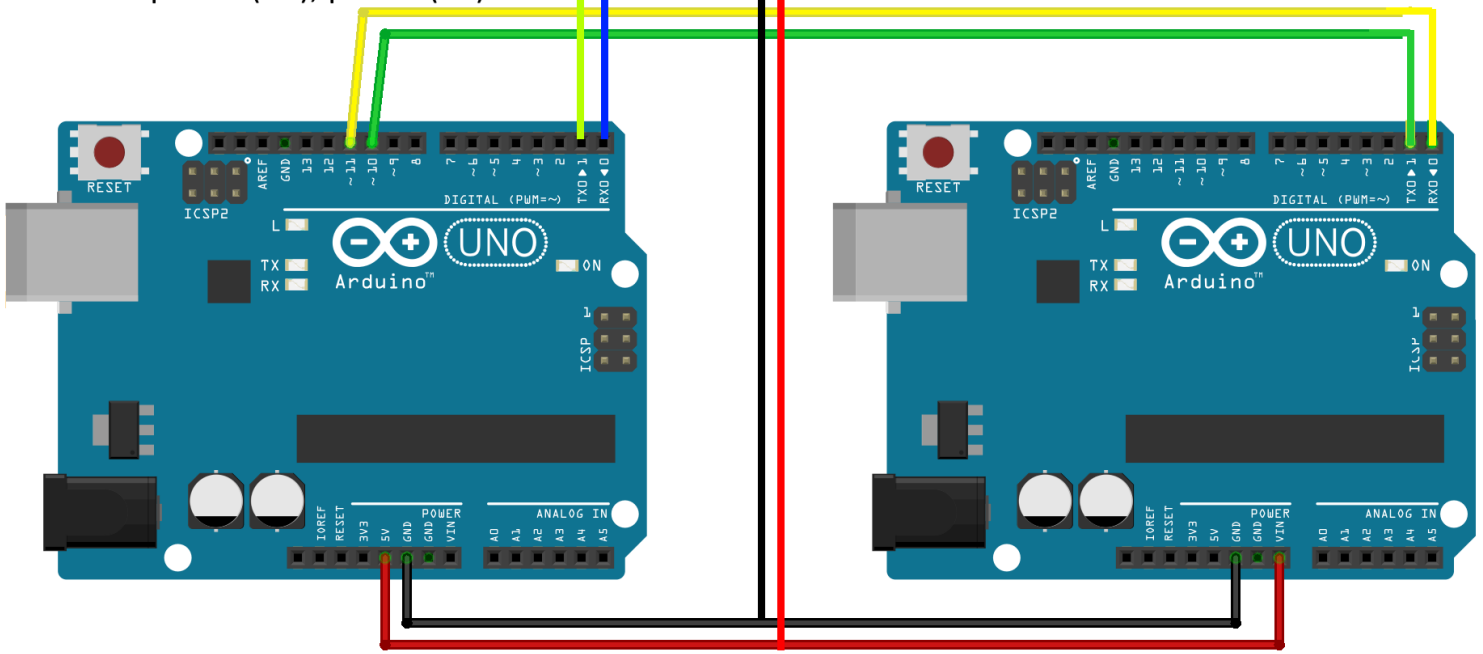
Replacing USB Cable with Bluetooth

Computer creates a new COM port for Bluetooth communication



Bluetooth Uses Hardware Serial pin 0 (RX), pin 1 (TX)

Software Serial pin 10 (RX), pin 11 (TX)



Additional Notes

- Arduino code stays the same when changing from USB to Bluetooth.
- How are the Arduino's powered? If powering Arduino with a 12V battery, then connect to the Vin pin (or Plug), and the Arduino board's linear regulator will step down the voltage to 5V.
- Ensure grounding pins are maintained across power supplies and between the two Arduino's.
- If your robot is turned on and off frequently, the Bluetooth will need to reconnect each time. Consider powering Bluetooth with a separate battery and 5V linear regulator to maintain the connection.
- MATLAB communication is the same as for USB (see lab 4), just have to find the COM port created for the Bluetooth.

MATLAB Sample Code

MATLAB Communication over Serial (i.e. COM Port)

```
serialInfo = instrhwinfo('serial');
comPorts = serialInfo.AvailableSerialPorts;

s = serial('COM4'); % creates serial object
fopen(s); % connects to serial object
pause(0.2);
```

repeat

```
fwrite(s,'r'); % writes letter to Arduino
sample = fscanf(s); % reads from Arduino
disp(sample);
```

```
fclose(s);   % if you don't close COM
             port after finishing you will
             need to restart MATLAB
```

Alternatively, use a loop

```
str = input('Input: ','s');
while str ~= 'q'

   fwrite(s,str); % writes letter to Arduino
   pause(0.5);
   while s.bytesAvailable > 0  % recieve data from Arduino
      sample = fscanf(s); % reads from Arduino
   end
   disp(sample);

   str = input('Input: ','s');
end
```