Sociable Objects Workshop

Instructor: Rob Faludi

Plan for Today

- Basic Pair Exercise: review
- Math for Mesh
- firmware updates if needed
- ZigBee and Arduino
- breakout board hookups
- doorbell exercises
- Readings & Assignments

Pairs Exercise Review

Math for Mesh

Math for Mesh

- Binary, Decimal, Octal, Hexadecimal
  - Why?
    - Serial communication
    - XBee configuration
    - Programming helper
  - What?
    - It’s all notation

Decimal

- Place system
- Powers
- Adding and carries
- Finger counting, but is that base 10?

Binary

- Place system
- Notation: %010 010b 0b10
- Powers
- Adding and carries
- Finger counting!

Octal

- Place system
- Notation **0**73
- Powers
- Adding and carries
- Finger counting, not really

Hexadecimal

- Place system
- Notation: extra digits, 0x10, #FFFFFF
- Powers
- Adding and carries
- Finger counting?
- Switches yes: 0xFF = 1111 1111 and 0x3C = 0011 1100

ASCII

- American Standard Code for Information Interchange
- 65 = A
- 48 = character zero, 49 = character one
- 32 = space, 10 = line feed, 13 = carriage return

One Question Quiz

- what is this: 10

Firmware Updates

Protocols

- Sending
- Flow control
- Call / response
- Broadcast
- Start / stop
- Checksums
- Collisions

ZigBee and Arduino

Why Arduino

- local logic
- pinouts
- fast prototyping
- lots of connection options

Arduino Serial Library

- Serial.begin(speed)
- Serial.available()
- Serial.read()
- Serial.flush()
- Serial.print(data)

Software Serial (internal)

- Hardware vs. software serial
  - 9600 baud max, typically pins 6 & 7 but any digital pins are okay
  - SWserial.read() is blocking
  - No serial.available() function in software serial
  - No buffering
  - Last choice for input, great for debug output w/ USB-serial converter
- http://www.arduino.cc/en/Reference/SoftwareSerial

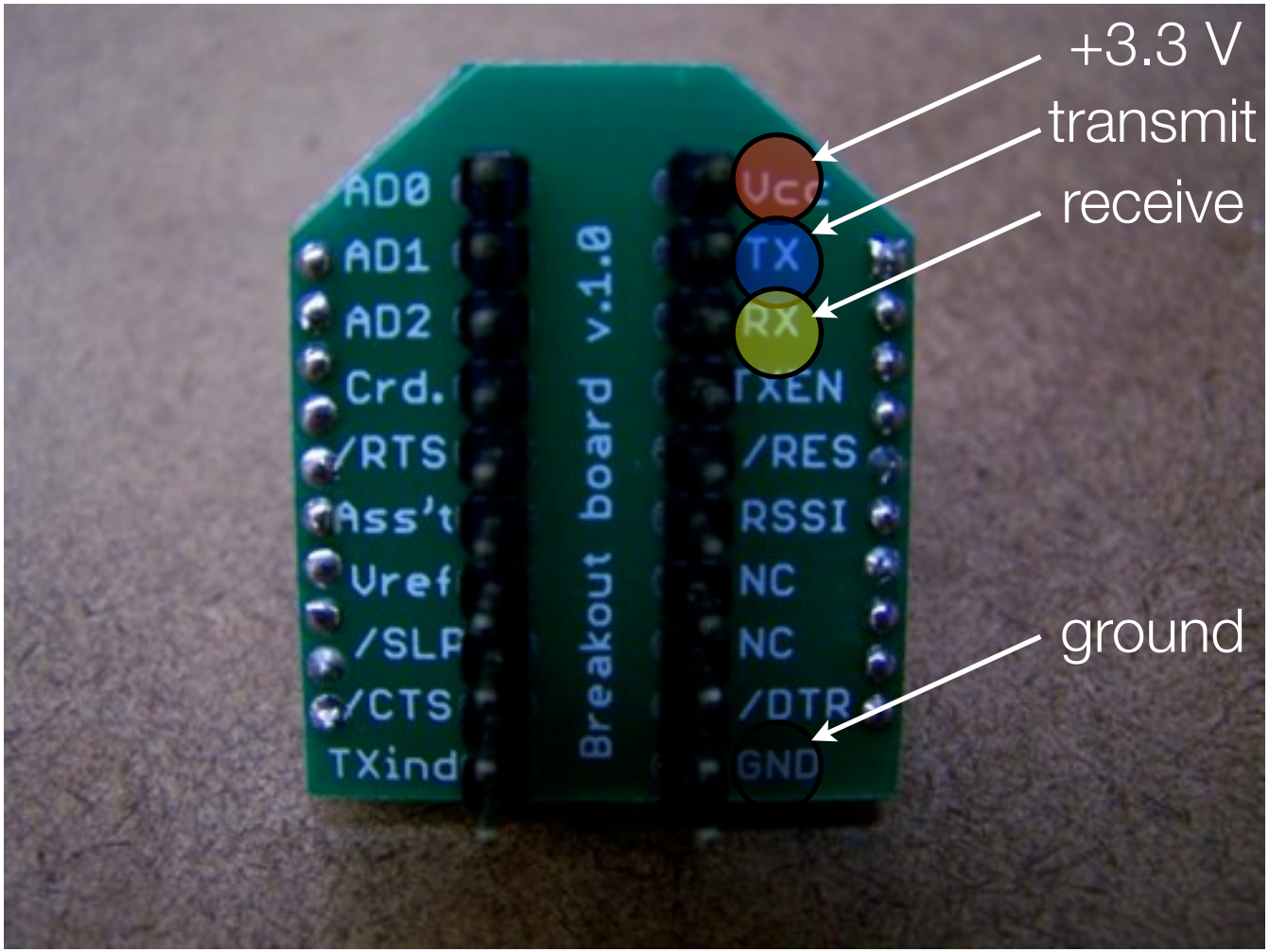
NEW Software Serial

- 115K baud max, all pins are okay to use
- SWserial.read() does not block (pretty sure about that...)
- serial.available() function in software serial
- buffering!
- good choice for input, great for debug output w/ USB-serial converter

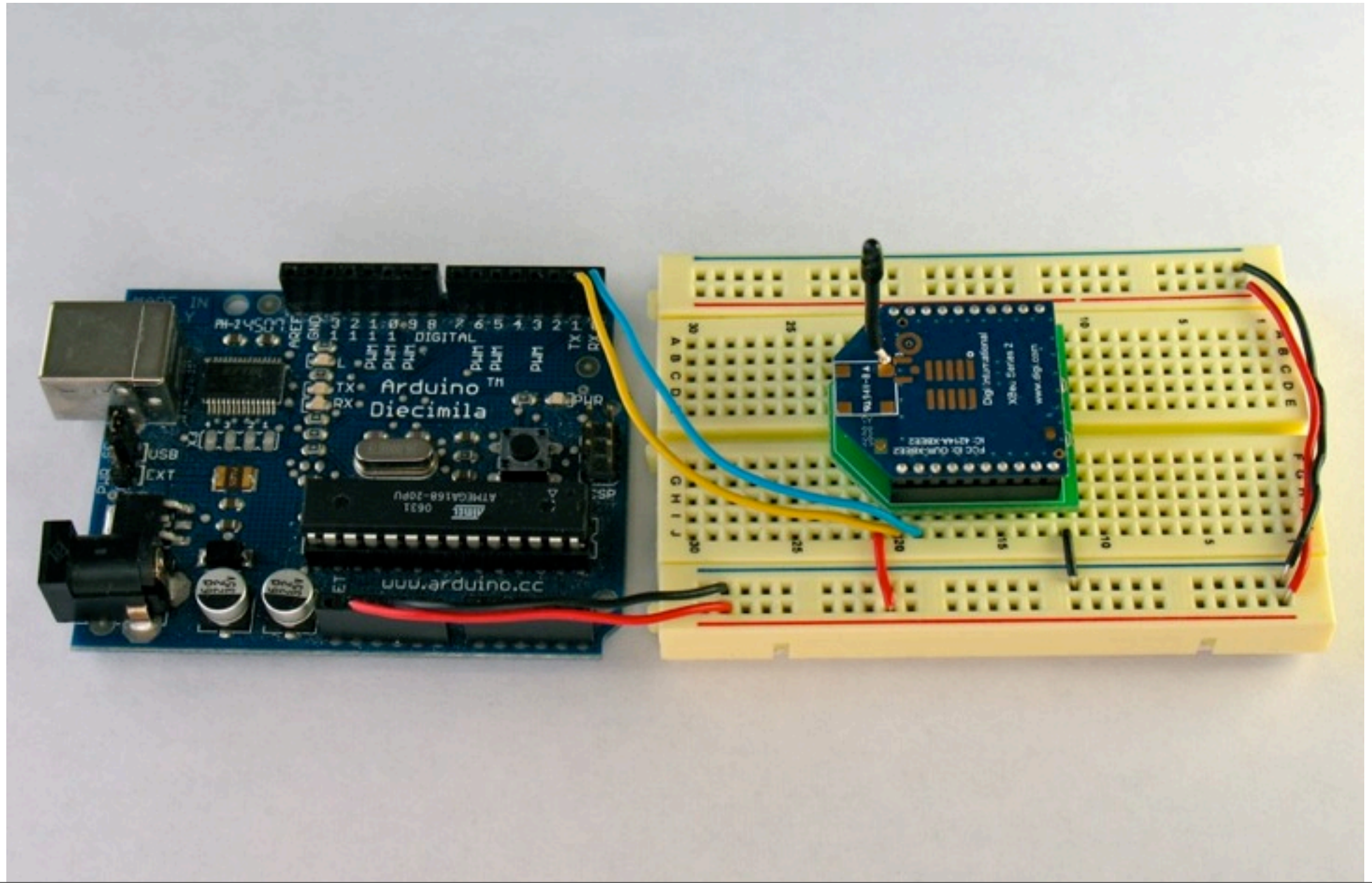
- http://arduiniana.org/libraries/NewSoftSerial/

Breadboard Hookups

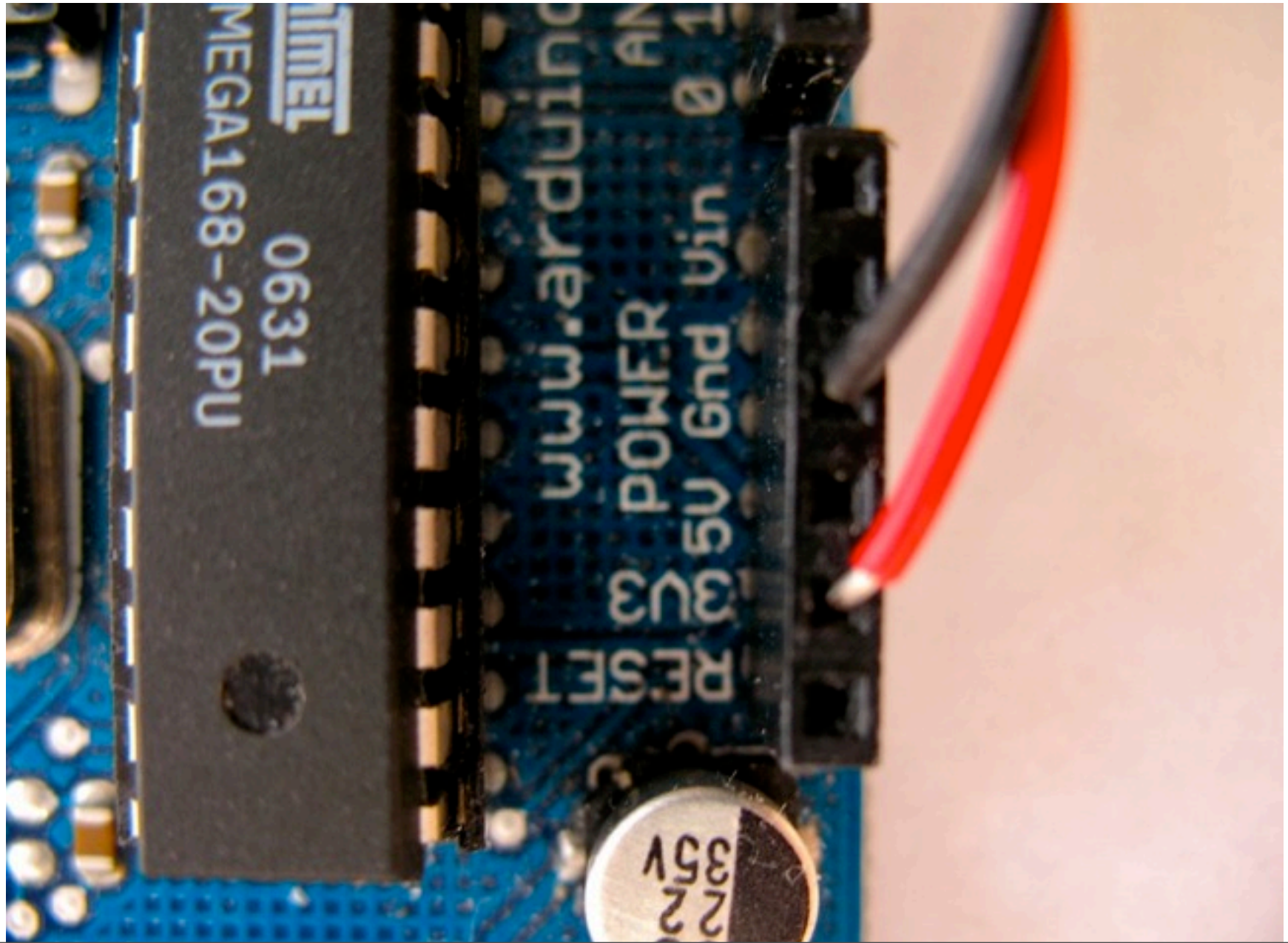
Wiring


+3.3 V
transmit
receive
ground
AD0
AD1
AD2
Crd.
/RTS
Ass't
Vref
/SLP
/CTS
TXind
Breakout board v.1.0
Vcc
TX
RX
TXEN
/RES
RSSI
NC
NC
/DTR
GND

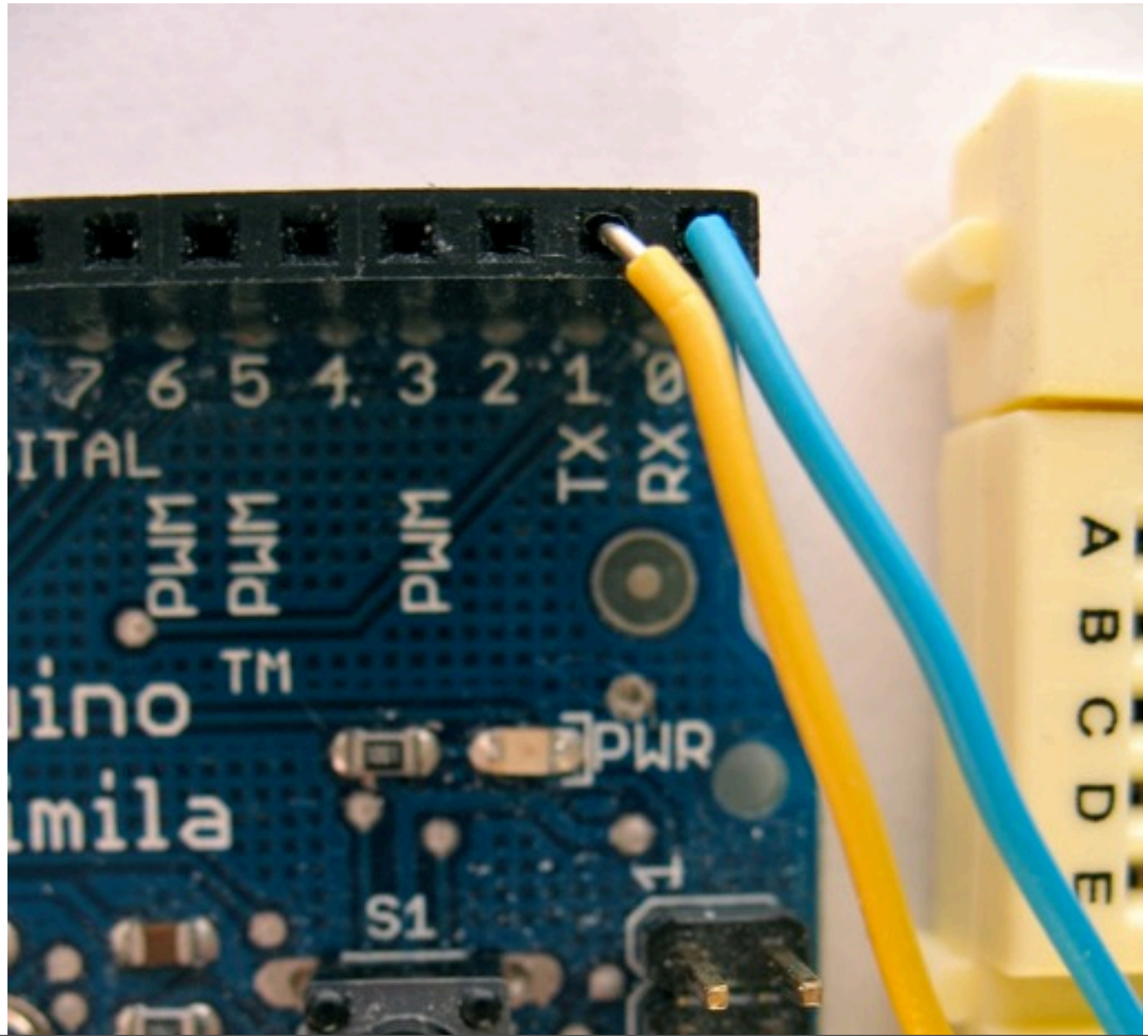
XBee Arduino Breadboard Layout



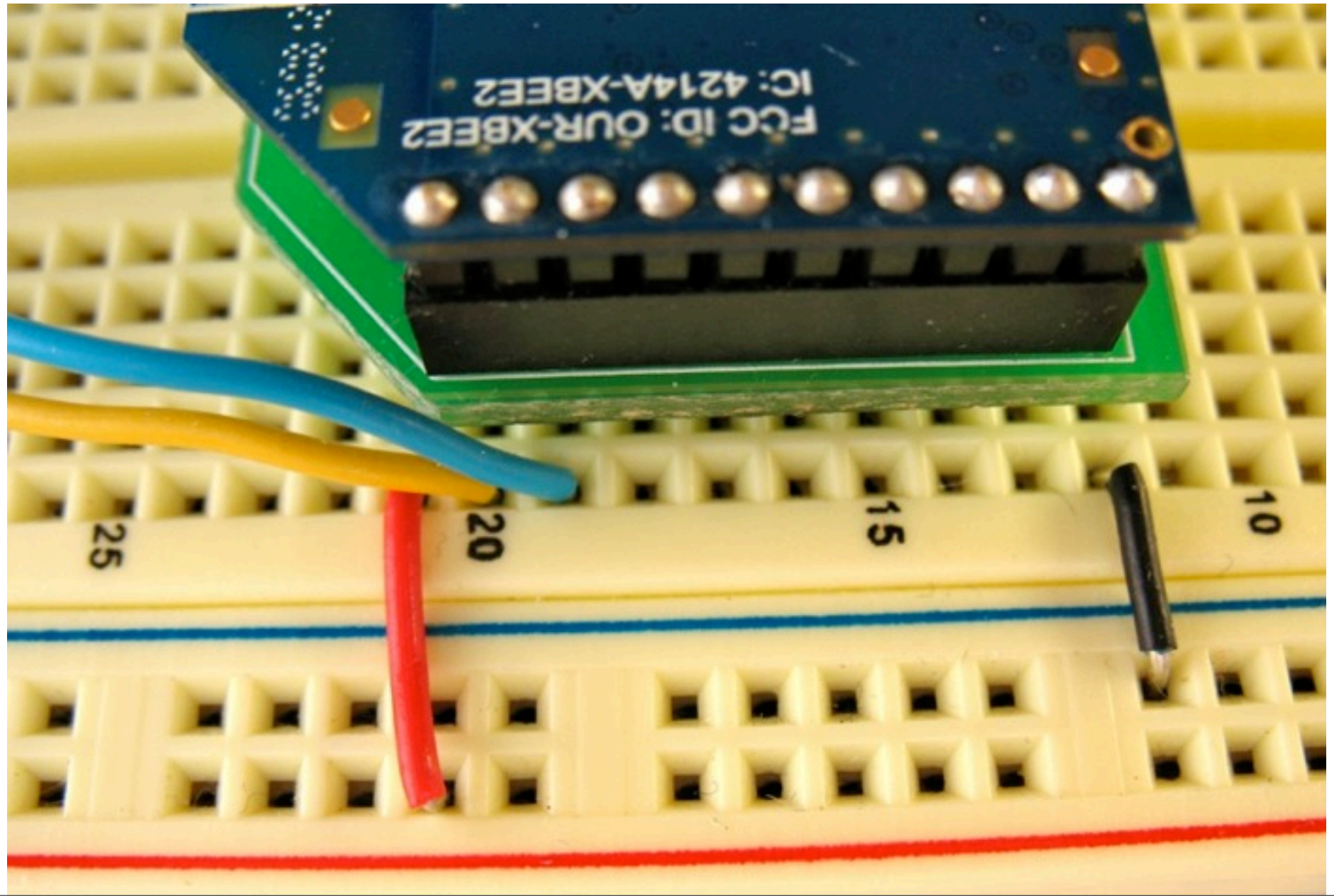
Power, Ground



TX, RX



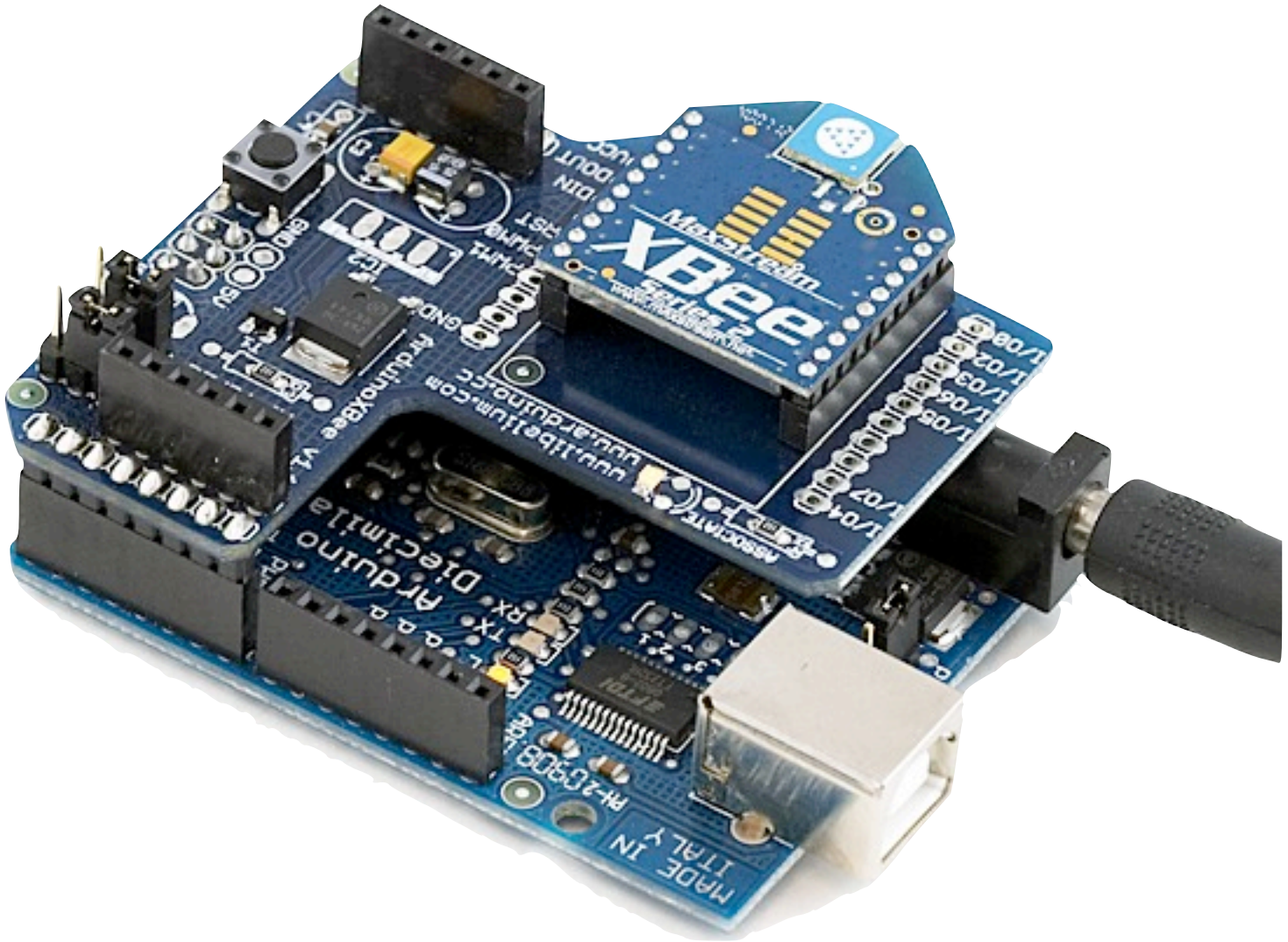
XBee Connections (pin 1, 2, 3 and 10)



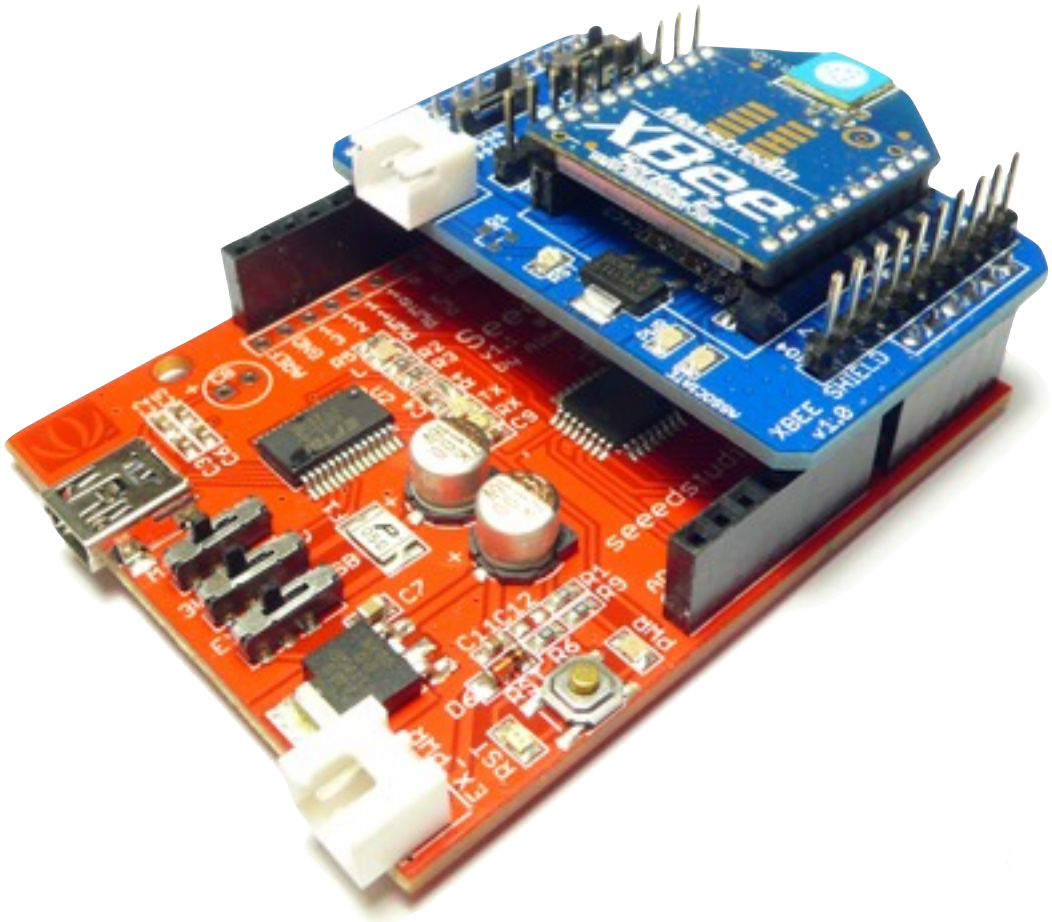
Remember!

- Use only +3.3 Volts. More than +7 Volts will kill your radio
- If you use a voltage regulator, always use decoupling capacitors. The radios often don't work without them.
- XBee TX goes to Arduino RX and vice versa.
- Unplug the TX & RX before uploading Arduino code (or use switches)
- You can't send infinitely fast. Try putting a 10 ms delay into your loop.

Arduino Shield



Seeduino Shield

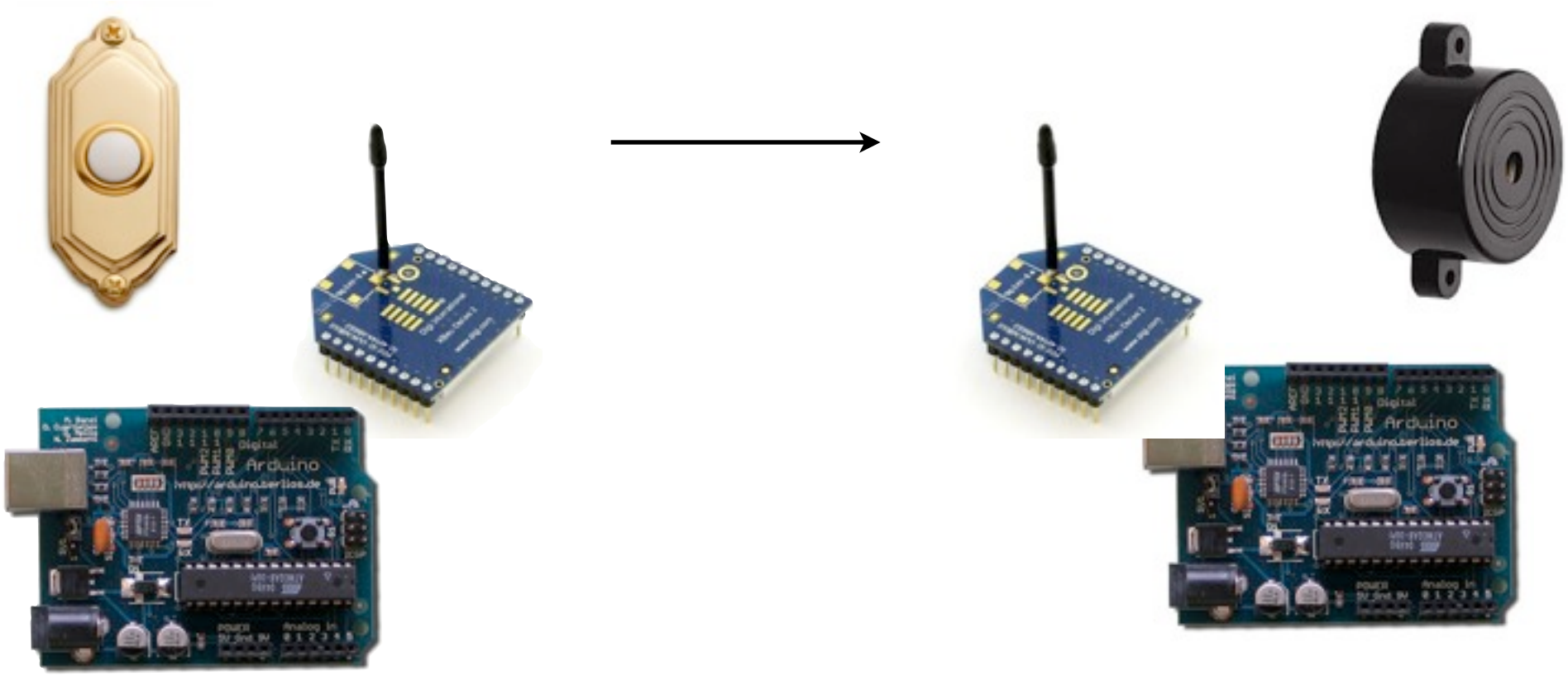


Doorbell Exercises

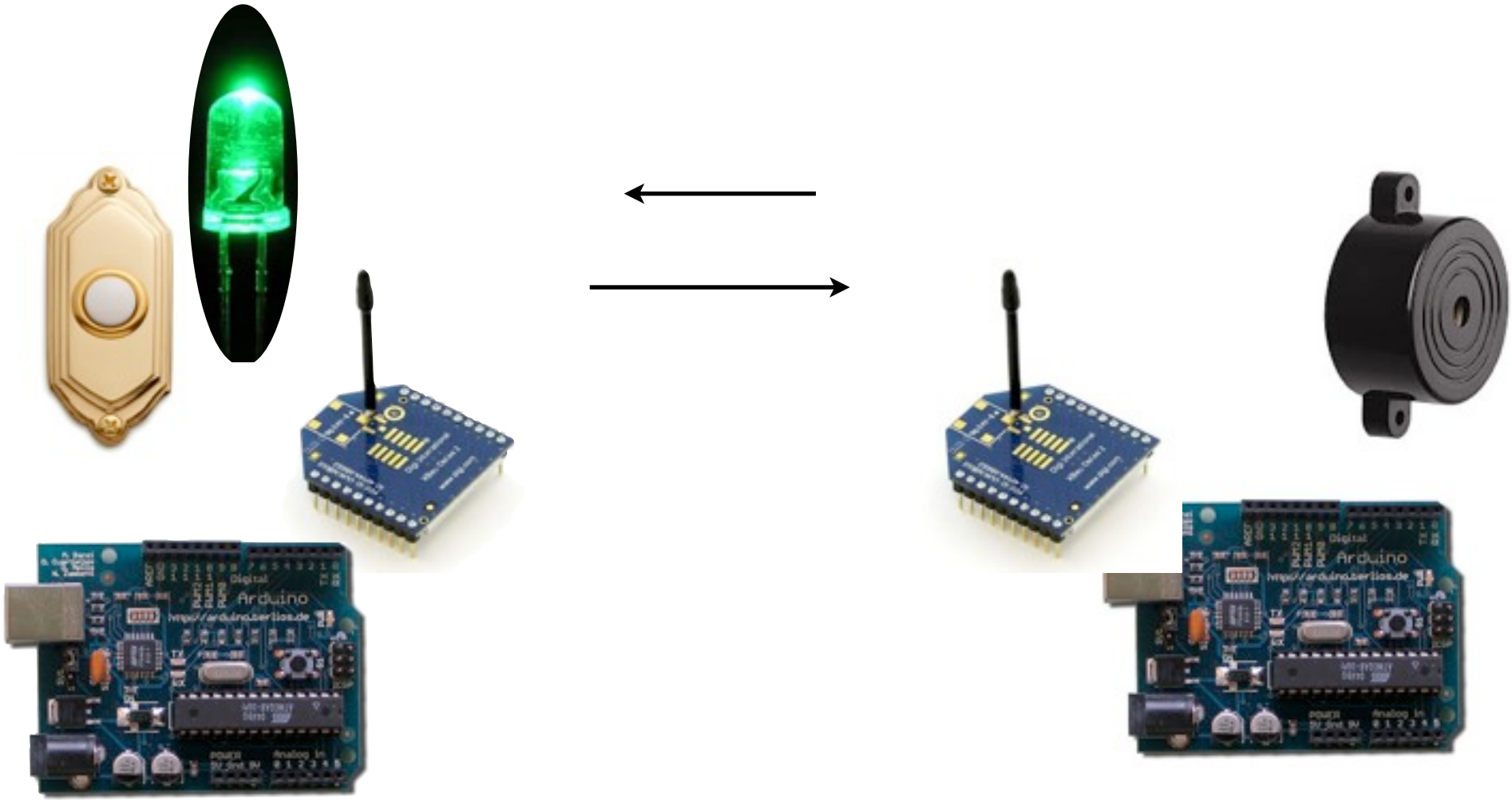
Remember

- All these exercises assume you have paired two radios together:
    - ATID set to the same value
    - ATDH & ATDL set to the SH and SL of the other radio
    - ATWR to write these settings so they're set for next power up
    - and don't forget that sometimes ATND will help recover your network
        - you can also try writing ATJV1 on the router so it resets automatically if it can't find its coordinator; useful if you change radios

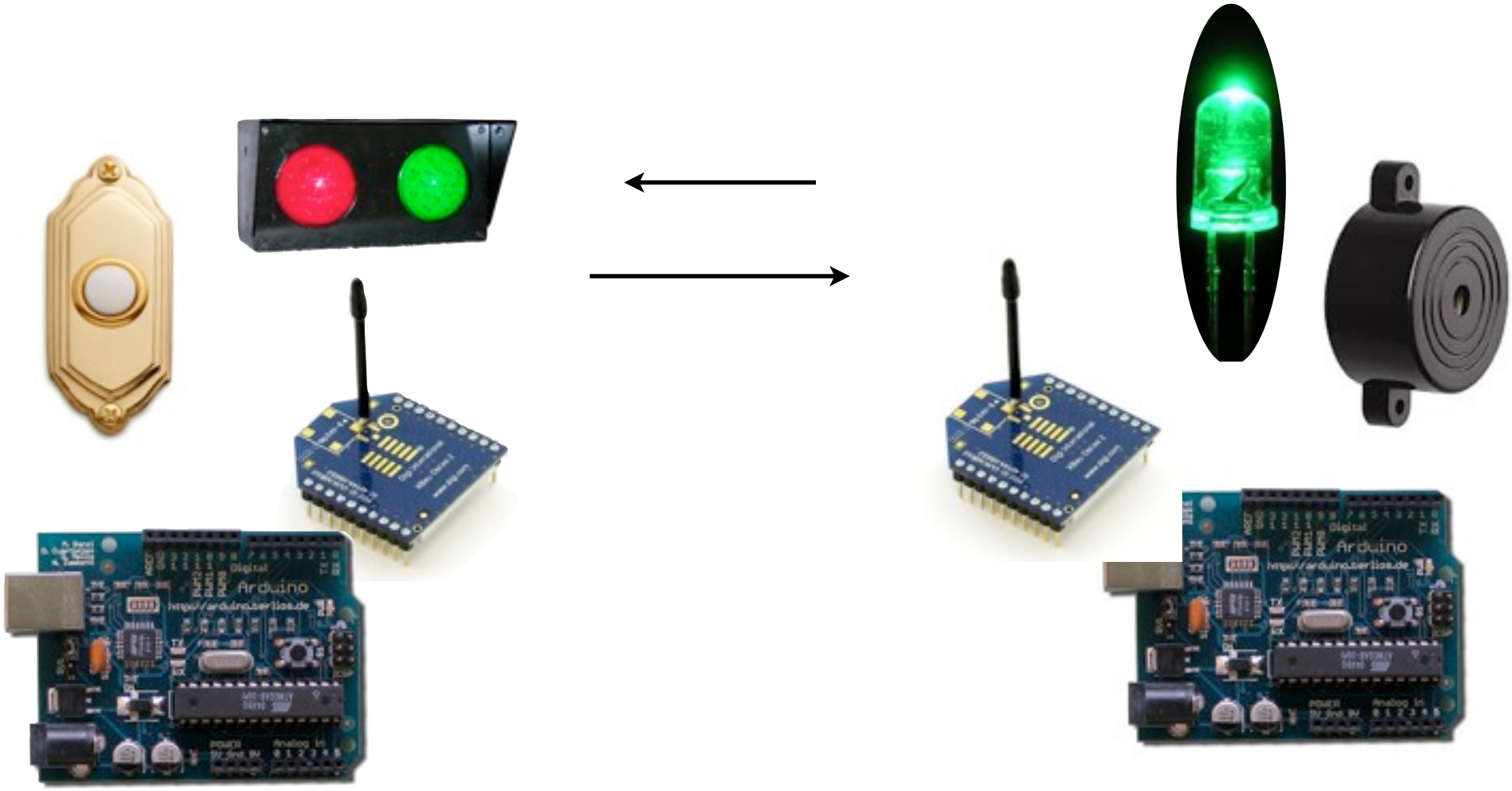
Basic Doorbell



Feedback Doorbell



Nap Doorbell



Other Doorbell Enhancements

Readings and Assignments

- Readings
    - Faludi Chapter 4
- Assignments
    - doorbell exercises
        - basic, feedback, nap, your own doorbells