creativity & computation lab

week 9 || talking: serial communication

review

WHERE WE HAVE BEEN

What we have done:

Sensors!

// Overview of different inputs

Actuators!

// Overview of outputs

Group work

Group presentations

agenda

WHERE WE ARE GOING

What's on for today:

millis()

// a new way to keep time

Arrays

// you can never get away from them!

Serial communication

// arduino --> processing

// processing --> arduino

Group presentations

last assignment

PRESENT

Show us your machine!

time

HOW CAN WE KEEP TIME WITH ARDUINO?

delay() vs. millis()

Pauses the program
for the amount of time
(in milliseconds)

time

HOW CAN WE KEEP TIME WITH ARDUINO?

delay() vs. millis()

Pauses the program for the amount of time (in milliseconds)

Returns the number of milliseconds since the Arduino board began running the current program.

millis()

AN EXAMPLE

millis()

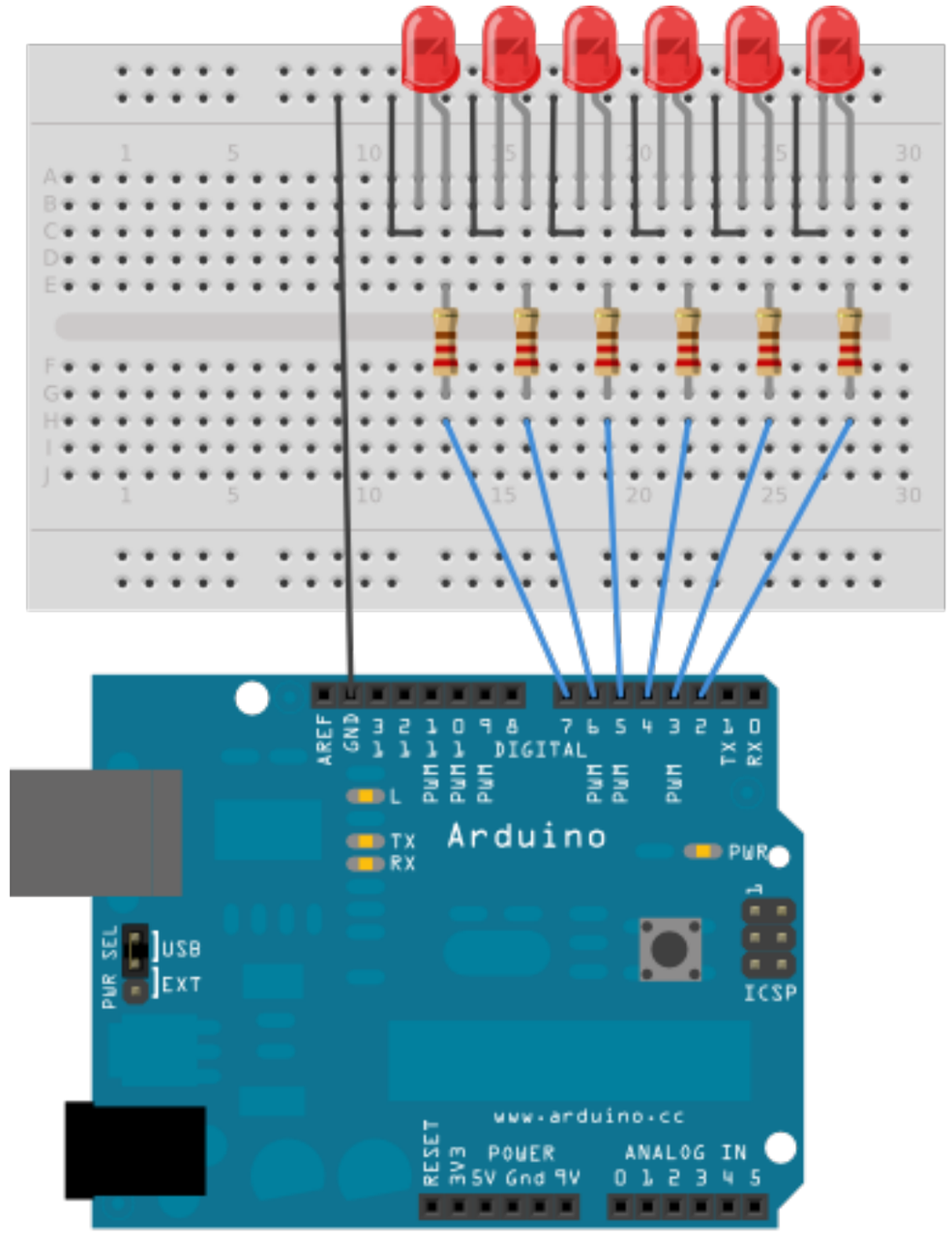
Let’s look at an example.

```
//In this example we are going to use variables to hold
the length of time since we turned an LED on or off.
```

arrays

YOU CAN USE THEM IN ARDUINO TOO

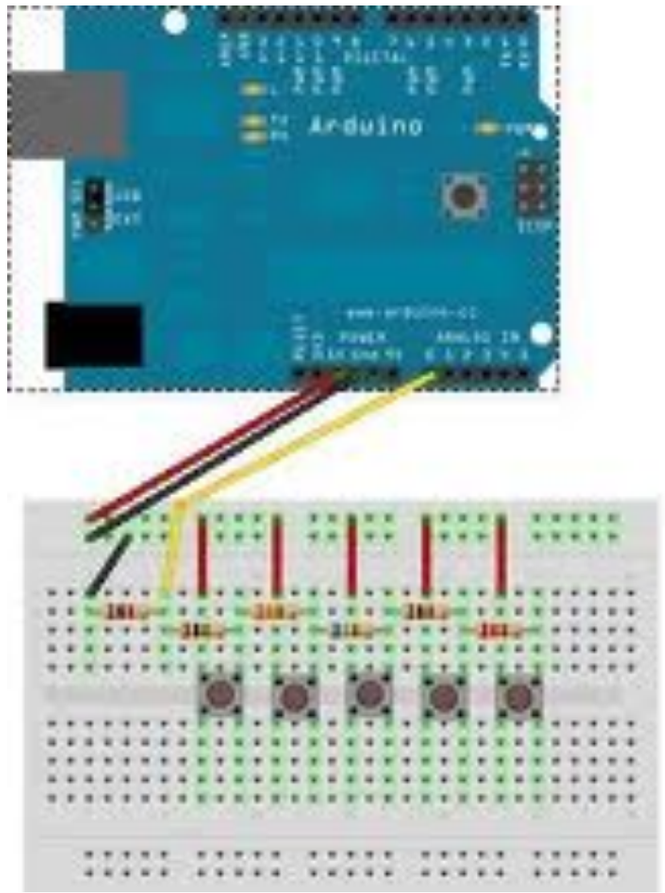
For output,
say if you want
to control
lots of LEDs



arrays

YOU CAN USE THEM IN ARDUINO TOO

For input,
say if you want
to control
lots of buttons



arrays

SYNTAX

Declare, but not initialize
```
int myInts[6];
```

Declare without choosing a size
```
int myPins[] = {2,4,8,3,6};
```

Initialize and size
```
int mySensVals[6] = {2,4,-8,3,2};
```

char message[6] = "hello";
```
char message[6] = "hello";
```

arrays

SYNTAX

Declare, but not initialize

```
int myInts[6];
```

Declare without choosing a size

```
int myPins[] = {2,4,8,3,6};
```

Initialize and size

```
int mySensVals[6] = {2,4,-8,3,2};
```

char message[6] = "hello";

```
char message[6] = "hello";
```

arrays

SYNTAX

Declare, but not initialize
```
int myInts[6];
```

Declare without choosing a size
```
int myPins[] = {2,4,8,3,6};
```

Initialize and size
```
int mySensVals[6] = {2,4,-8,3,2};
```

char message[6] = "hello";
```
char message[6] = "hello";
```

arrays

SYNTAX

Declare, but not initialize

```
int myInts[6];
```

Declare without choosing a size

```
int myPins[] = {2,4,8,3,6};
```

Initialize and size

```
int mySensVals[6] = {2,4,-8,3,2};
```

char message[6] = "hello";

```
char message[6] = "hello";
```

arrays

LET'S LOOK AT AN EXAMPLE

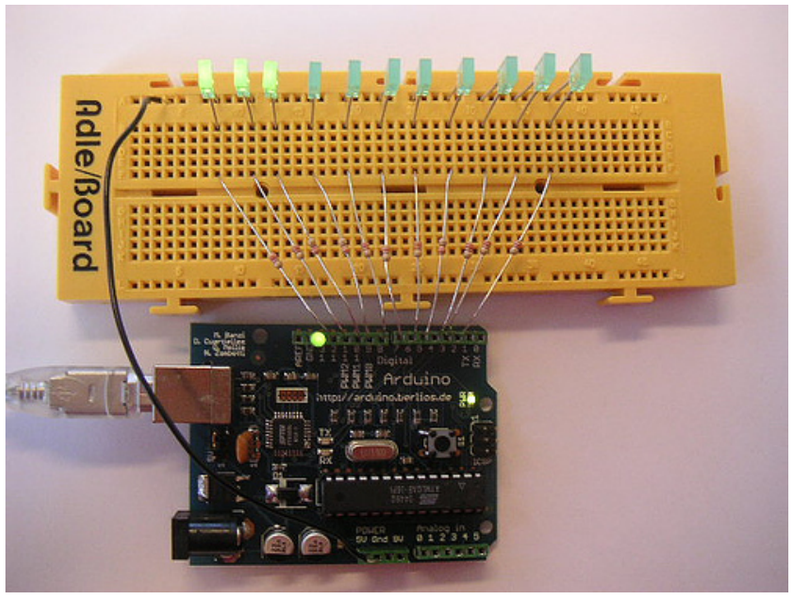
Examples >> Control >> Arrays

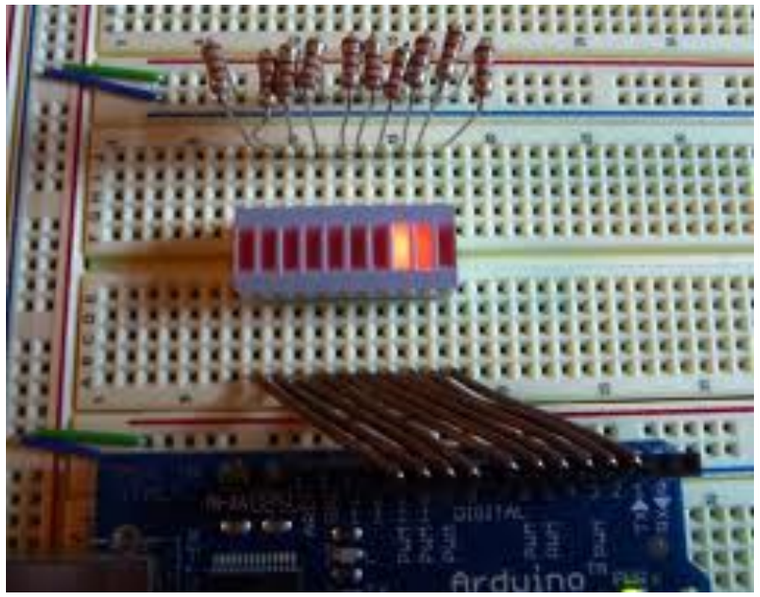
Arrays

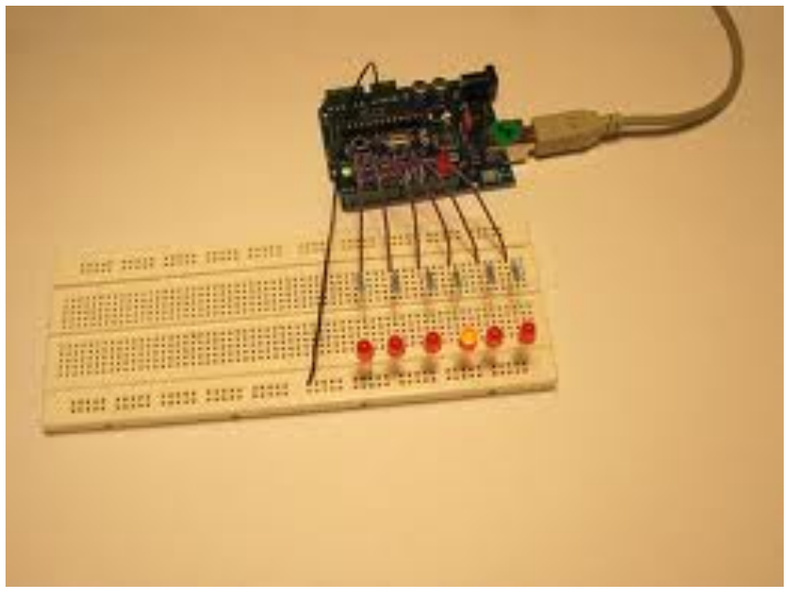
```
This example code is in the public domain.

 http://www.arduino.cc/en/Tutorial/Array
 */

int timer = 100;           // The higher the number, the slower the timing.
int ledPins[] = {
  2, 7, 4, 6, 5, 3 };       // an array of pin numbers to which LEDs are attached
int pinCount = 6;           // the number of pins (i.e. the length of the array)

void setup() {
  int thisPin;
  // the array elements are numbered from 0 to (pinCount - 1).
  // use a for loop to initialize each pin as an output:
  for (int thisPin = 0; thisPin < pinCount; thisPin++)  {
    pinMode(ledPins[thisPin], OUTPUT);
  }
}

void loop() {
  // loop from the lowest pin to the highest:
  for (int thisPin = 0; thisPin < pinCount; thisPin++) {
    // turn the pin on:
    digitalWrite(ledPins[thisPin], HIGH);
    delay(timer);
    // turn the pin off:
    digitalWrite(ledPins[thisPin], LOW);

  }

  // loop from the highest pin to the lowest:
  for (int thisPin = pinCount - 1; thisPin >= 0; thisPin--) {
```

exercise time!

THE KNIGHTRIDER EXAMPLE. CLASSIC.







http://www.youtube.com/watch?v=VgMpZFDnlEU

exercise time!

MAKE YOUR OWN.

You can work in groups and use the internet ONLY if you get stuck on something.

DO NOT copy/paste code.

If you finish early, try with multiple inputs!

communication

HOW DO COMPUTERS TALK TO EACH OTHER

Talking and listening

Need to agree upon who will talk first and who will listen first

Develop a protocol

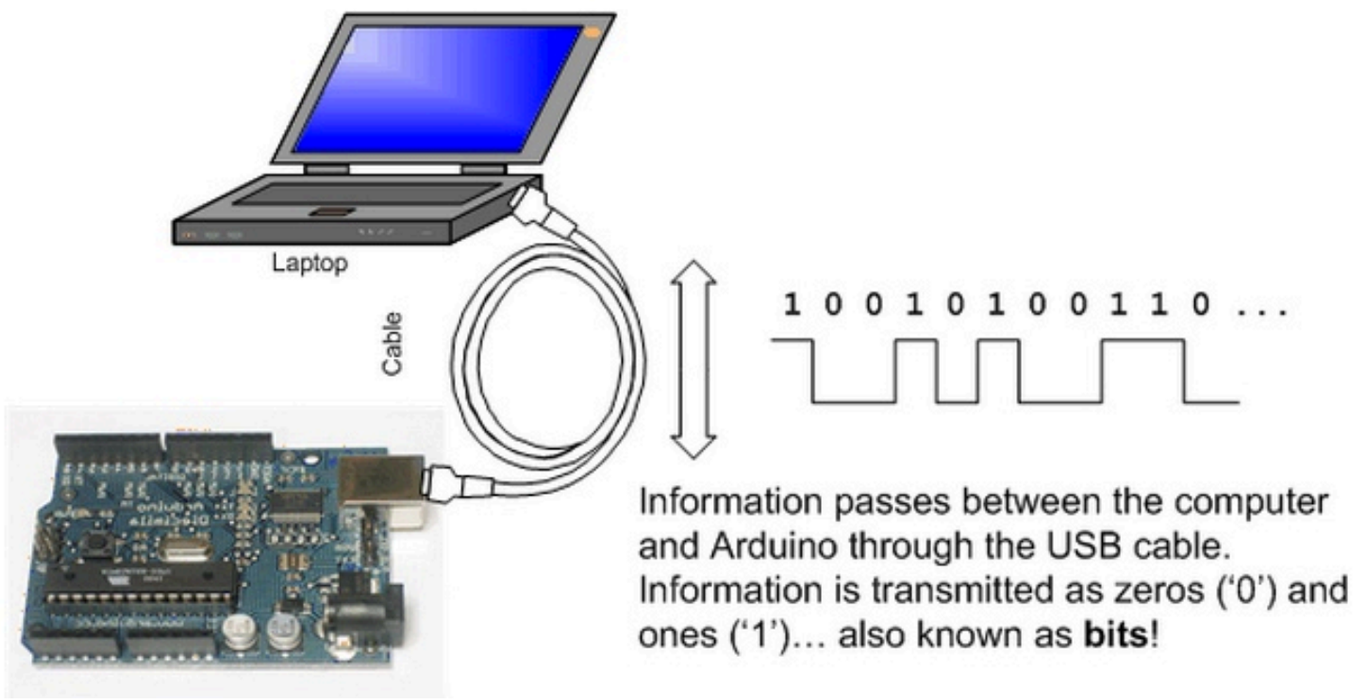
Many ways to set up communication
//Serial Communication, I2C, etc

serial communication

HOW DO COMPUTERS TALK TO EACH OTHER

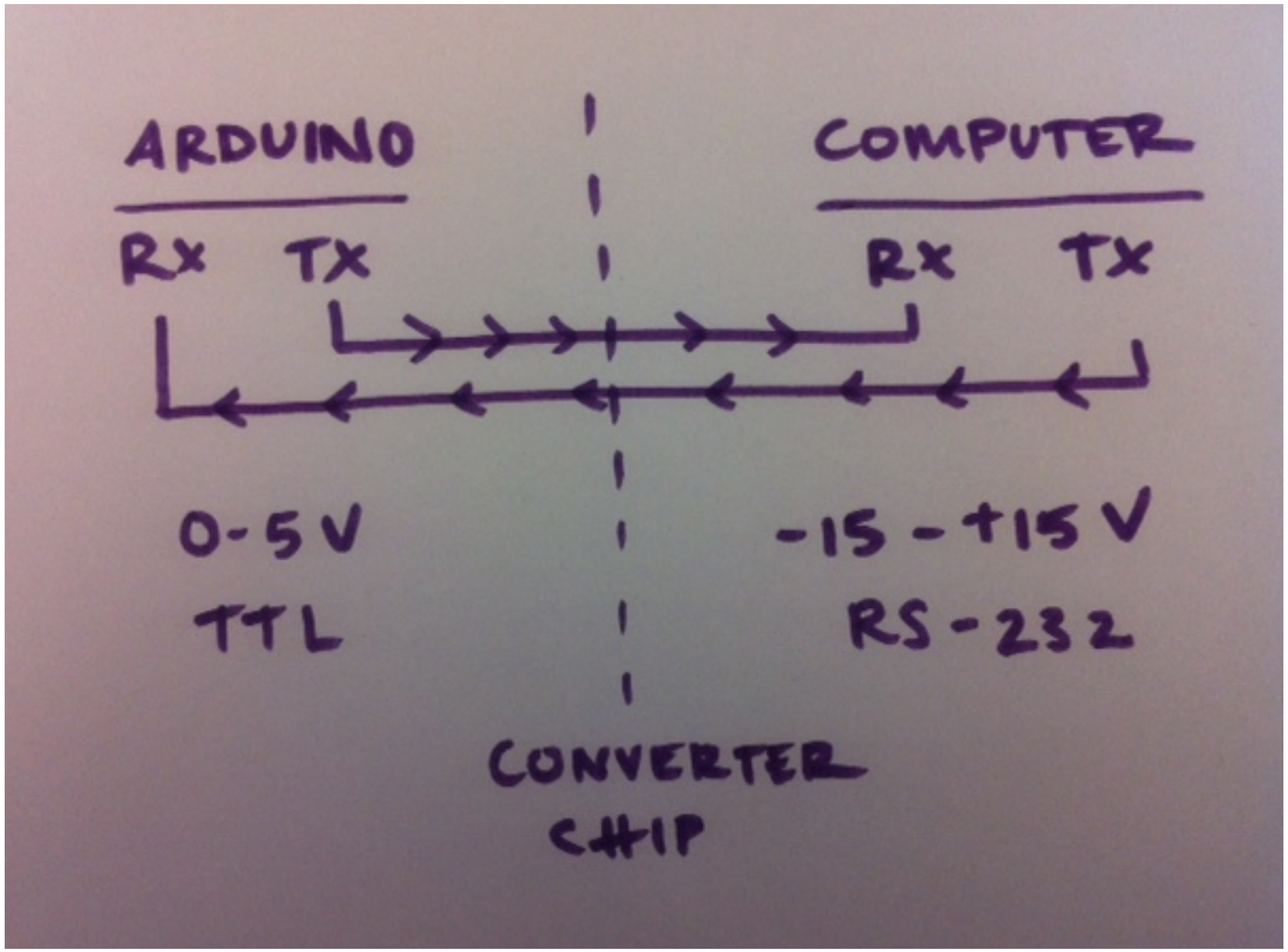
What is serial communication?

//One after another - sending bits and bytes of info



serial communication

HOW DO COMPUTERS TALK TO EACH OTHER



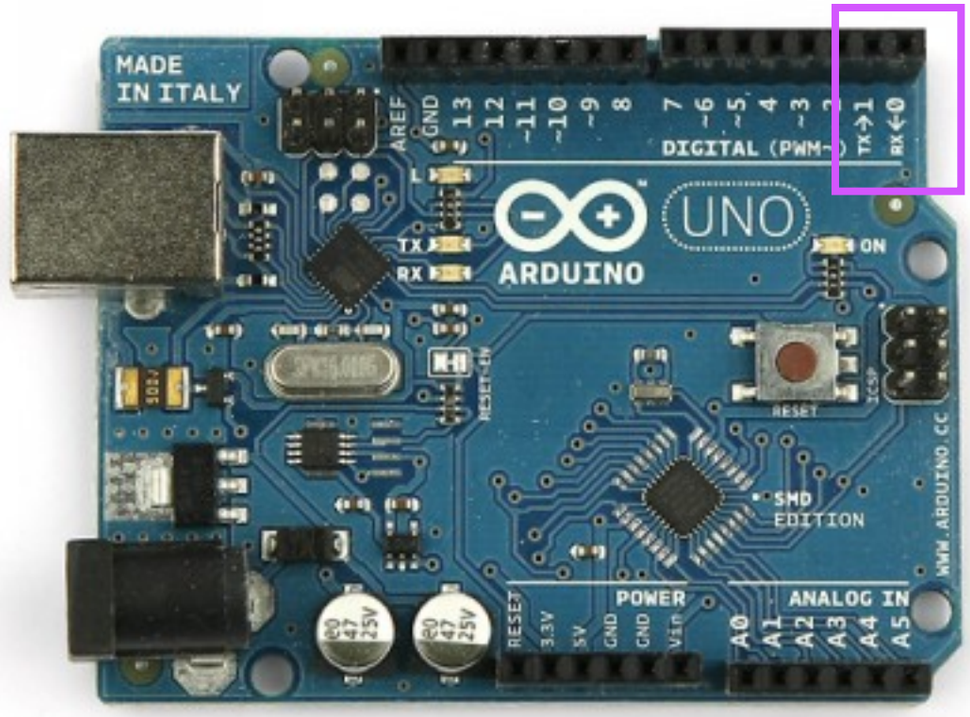
Original drawing from here: http://www.youtube.com/watch?v=g0pSfyXOXj8

serial communication

SO THERE'S A SPECIAL PLACE ON THE BOARD FOR THIS

TX and RX pins

//Function like regular input/ output pins until you use serial communication.



serial communication

HOW COMPUTERS TALK TO EACH OTHER

The Serial class in Arduino

Serial.begin() = setup the Serial library

Serial.println() = Prints data to the serial port as human-readable ASCII text followed by a carriage return character (ASCII 13, or '\r') and a newline character (ASCII 10, or '\n')

Serial.print() = Prints data to the serial port as human-readable ASCII text.

serial communication

HOW COMPUTERS TALK TO EACH OTHER

The Serial class in Arduino

Serial.available = bytes available for reading. Data that's already arrived and stored in the serial receive buffer

Serial.read() = reads incoming data

Serial.write() = writes binary data to the serial port

Serial.end() = disables serial communication, allowing the RX and TX pins to be used for general input and output

exercise time!

BECAUSE OUR BRAINS NEED A WORKOUT TOO

Let’s talk to the computer first through the serial monitor.

exercise time!

BECAUSE OUR BRAINS NEED A WORKOUT TOO

Let’s talk to the computer first through the serial monitor.

Now let’s have Arduino talk to Processing.

exercise time!

BECAUSE OUR BRAINS NEED A WORKOUT TOO

Let’s talk to the computer first through the serial monitor.

Now let’s have Arduino talk to Processing.

Now Processing to Arduino (this is all you).

midterm

WHAT YOU WILL BE DOING

Option A:

Iterate on an assignment that you wanted to take further.

Option B:

Propose something new. You should submit proposals to me via EMAIL before next class. You should include:

- _ A concept statement and the program you will use to realize it
- _ Why do you want to do this? How this will contribute to your learning of a particular area?

I will email/post further details later this week.

for next class

WHAT YOU WILL BE DOING

1) Option A: Iterate on your Rube Goldberg machine to include Processing.

Option B: Create a realtime data visualization in Processing using sensor input.

2) Prepare for a midterm project workshopping session that will take place during the second half of class. Bring your concept, prototypes, and problems!