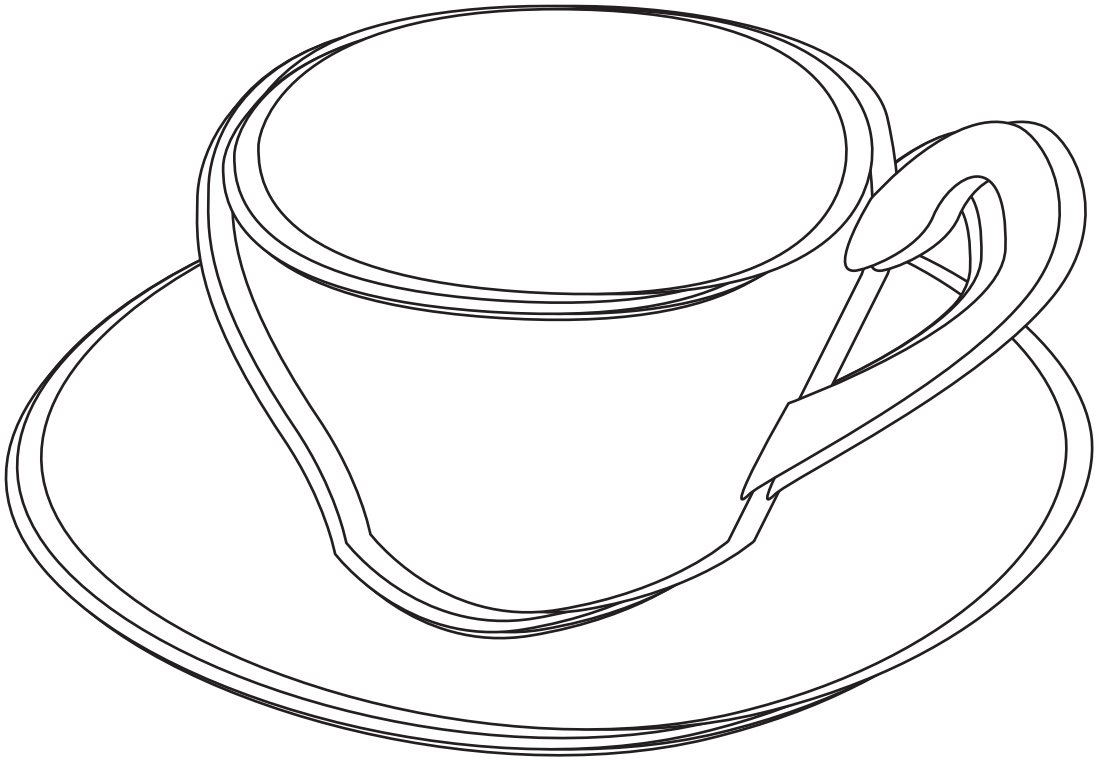
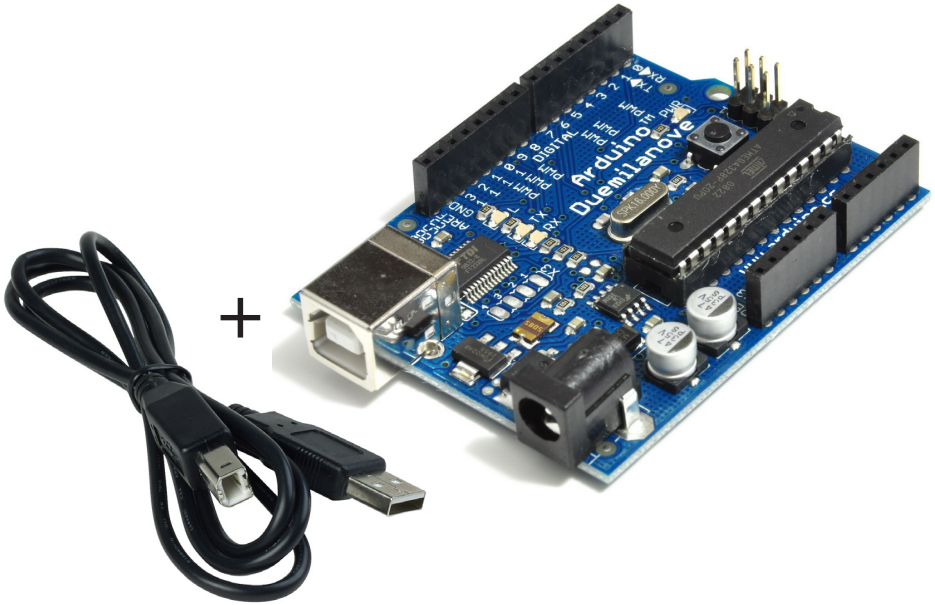
SHIVER MY TEACUP
BY Mztek and Katrin Baumgarten

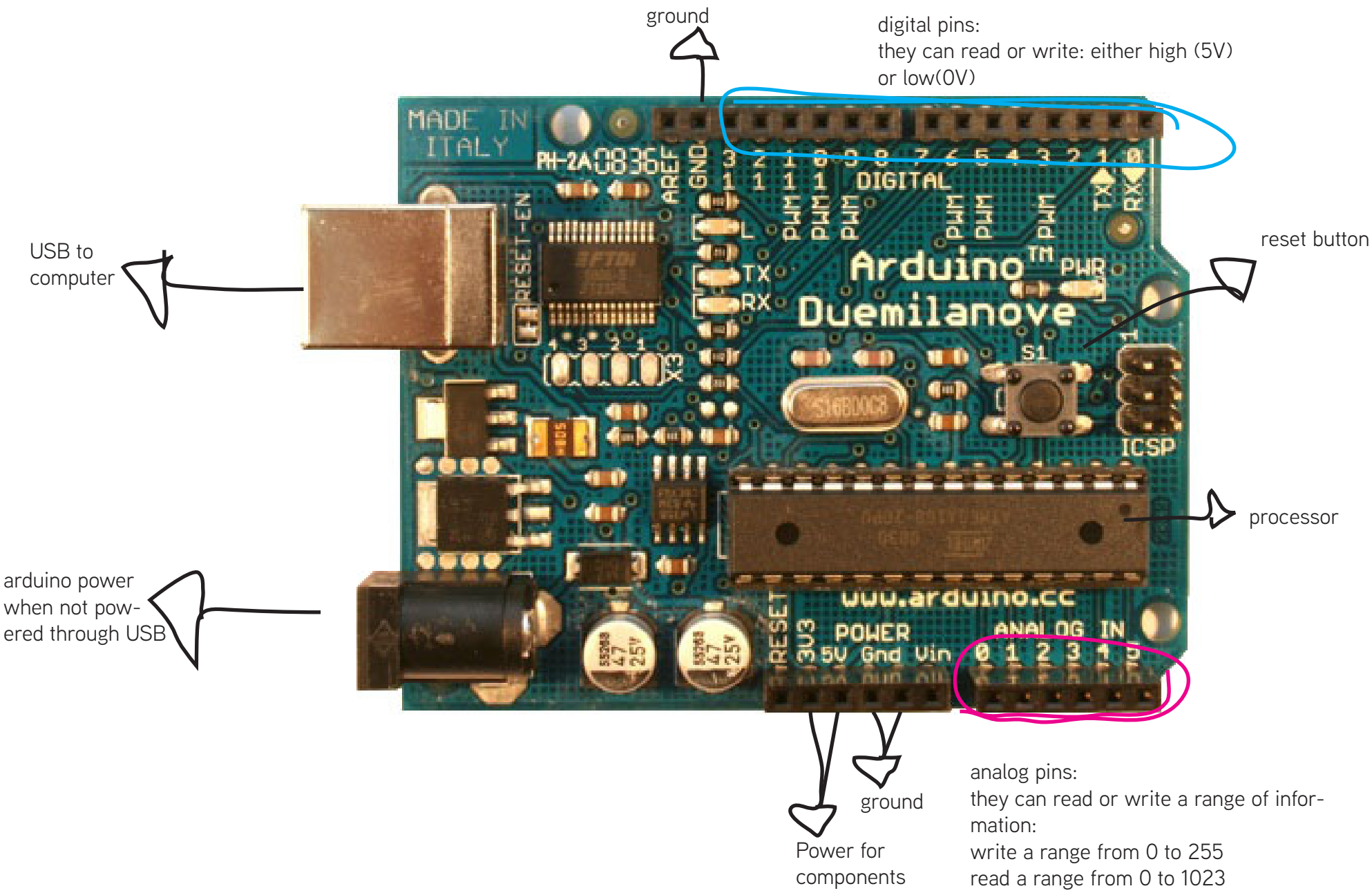


Tinkering is what happens when you try something you don't quite know how to do, guided by a whim, imagination and curiosity. When you tinker, there are no instructions, but there also no failures, no right or wrong way of doing things. It's about figuring out how things work and reworking them.

- Massimo Banzi, one of the originators of the Arduino project

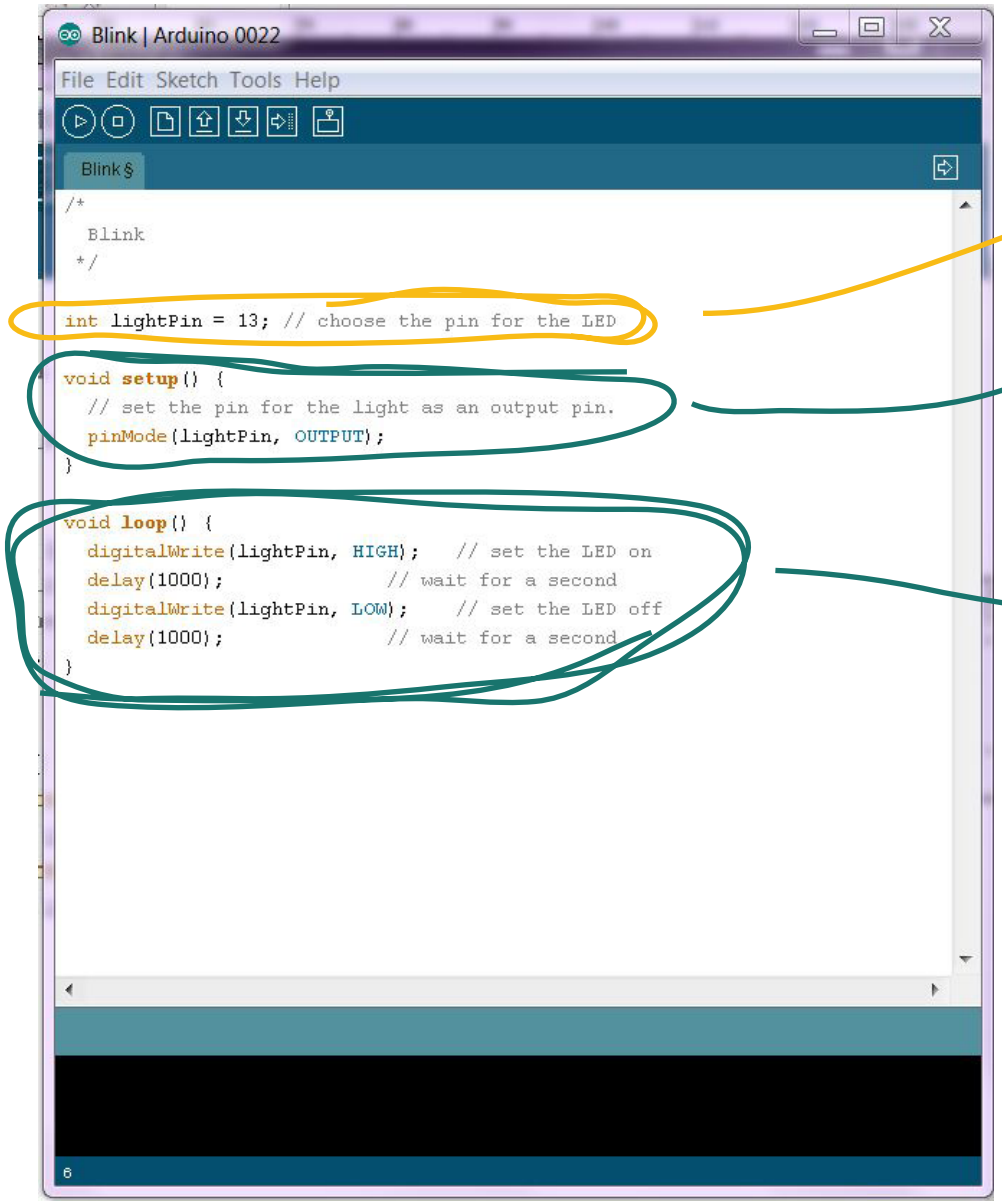


download and install program from here:
» http://www.arduino.cc/en/Guide/HomePage


ground
digital pins:
they can read or write: either high (5V)
or low(0V)
USB to
computer
reset button
processor
arduino power
when not pow-
ered through USB
ground
Power for
components
analog pins:
they can read or write a range of infor-
mation:
write a range from 0 to 255
read a range from 0 to 1023
MADE IN
ITALY
DIGITAL
Arduino
Duemilanove
PWR
ICSP
www.arduino.cc
POWER
ANALOG IN

ARDUINO BLINK EXAMPLE:

File > Examples > 1.Basics > Blink



initialization
these variables will be available throughout the code

setup
get everything ready for the program to run

loop
repeat action until told otherwise

PROGRAMMING ESSENTIALS:

datatypes (each variable has a type)

int	a number without a decimal point, for example 4 or -12
float	a number with a decimal point, for example 1.23 or -128.12
char	a single character or number that will be treated as number or character for example: a, 1, !
byte	The value of a byte, between -128 and 127 if the byte is signed and 0 and 255 if it's not signed
boolean	A true or false value

control statements

operators and their uses

+, -, *, /	adds, subtracts, multiplies, and divides
%	modulo; returns the divider of a division
=	assignment; assigns the value on the right to the variable on the left
+=, -=, *=, /=	mathematical assignment; adds, subtracts, multiplies, and divides the value on the left by the value on the right and sets the value on the right to that result
++	adds 1 to the value to the left
--	subtracts 1 from the value to the right
==	compares the value on the left with the value on the right. If they are equal, then expression is true.
!=	compares the value on the left with the value on the right. If they are not equal, then expression is true.
>, >=	compares the value on the left with the value on the right. If the value on the left is greater than or greater than or equal to the other, the expression is true.
<, <=	compares the value on the left with the value on the right. If the value on the left is lesser than or lesser than or equal to the other, the expression is true.
&&	Check the truth of the expression to the left of the operator and to the right, if both are true, the entire expression is true
| |	Check the expression to the left of the operator to the right, if either is true, the entire expression is true

more language references can be found here: http://www.arduino.cc/en/Reference/HomePage

PROGRAMMING ESSENTIALS:

control statements

```
if (condition) {
  result if the condition is true
}
else {
  result if the condition is false
}
```

if/then — conditional logic statement

```
for(i=0; i<10; i++) {
  print("i is" +i)
  }
```

for Loop — this statement lets us do things over and over again for a specified number of repetitions

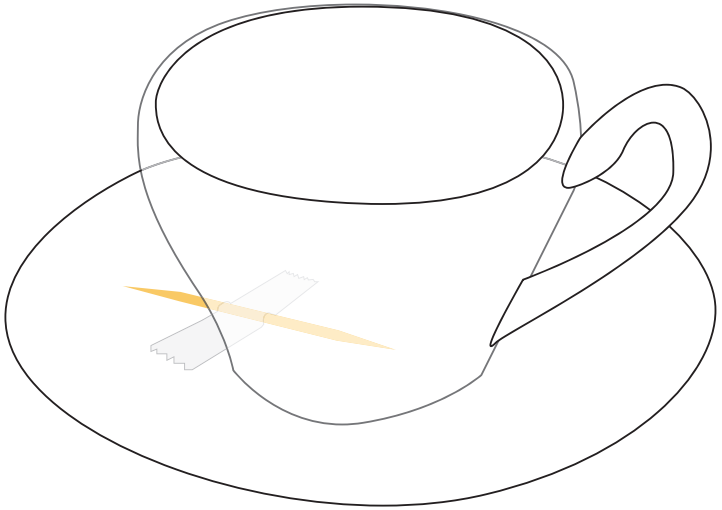
```
while (trueOrfalse){
  something to do each time
```

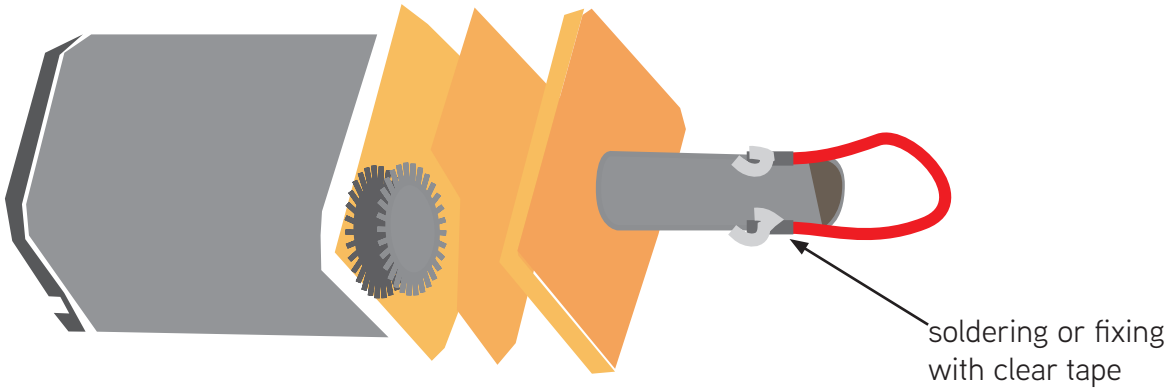
while Loop — is similar to the for loop, but slightly less sophisticated

functions

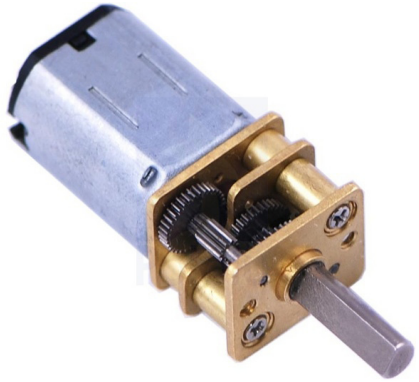
```
int functionName (int x){
doSomething;
}
```

A function is a name for a grouping of one or more lines of code and is somewhat like a variablein that it has type and a name. But it doesn't just store information; it manipulates it.



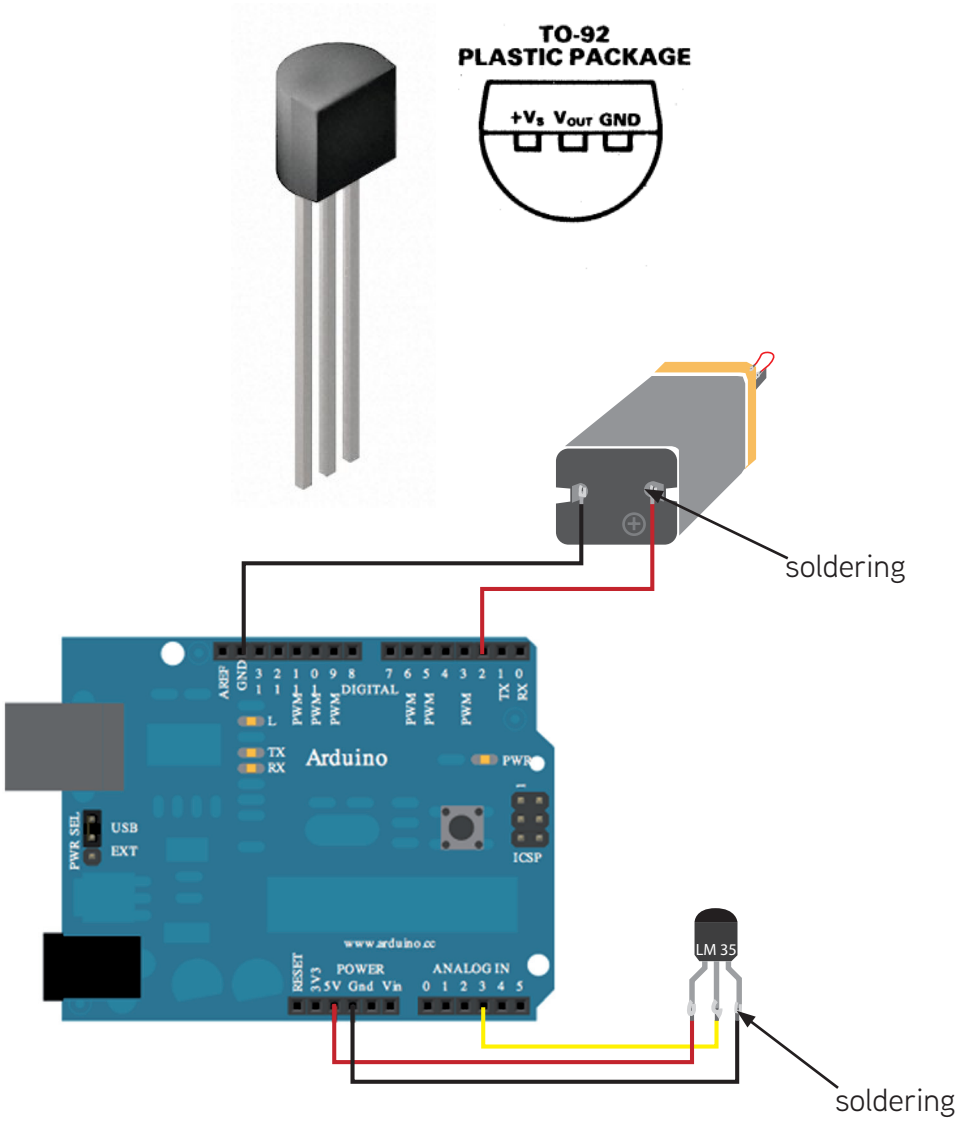


100:1 Micro Metal Gearmotor

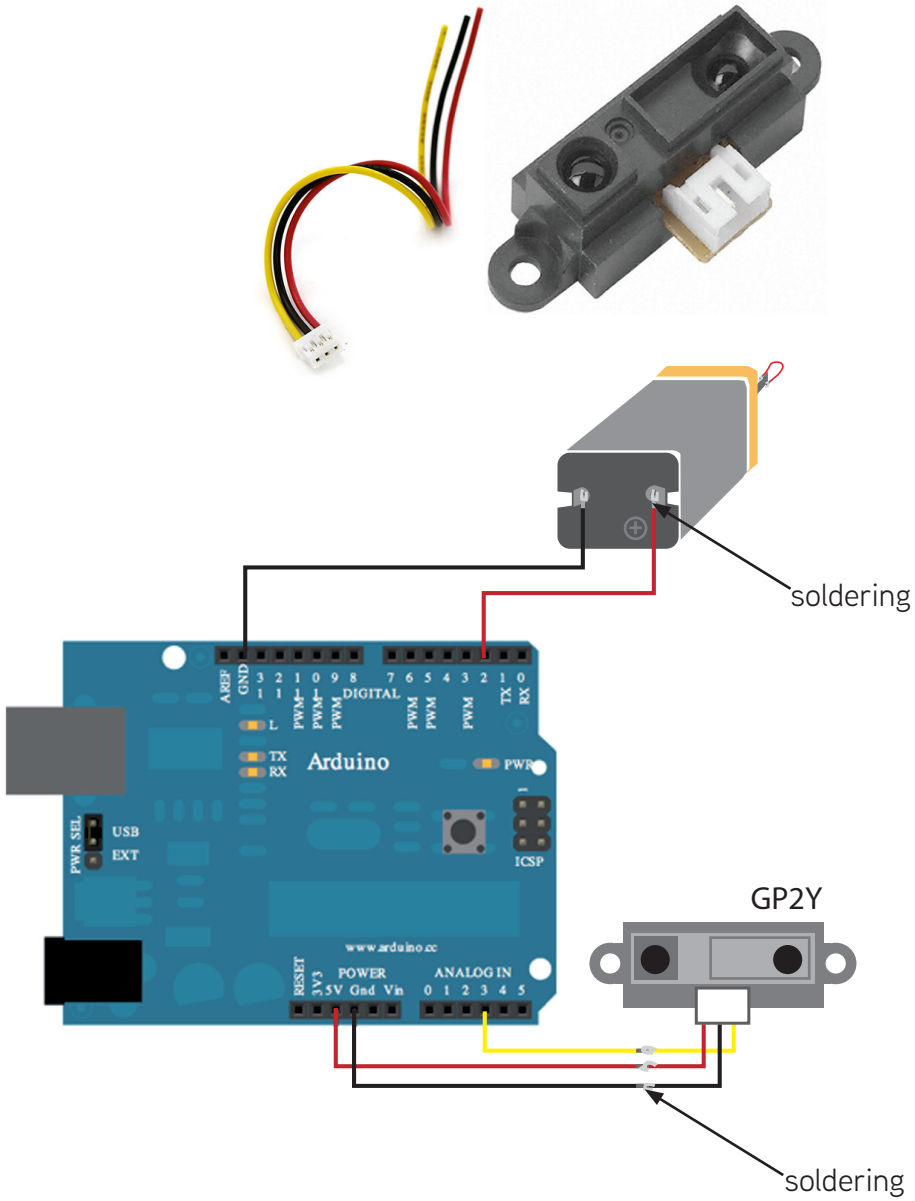


THE CIRCUIT
WHICH PART TO WHERE?

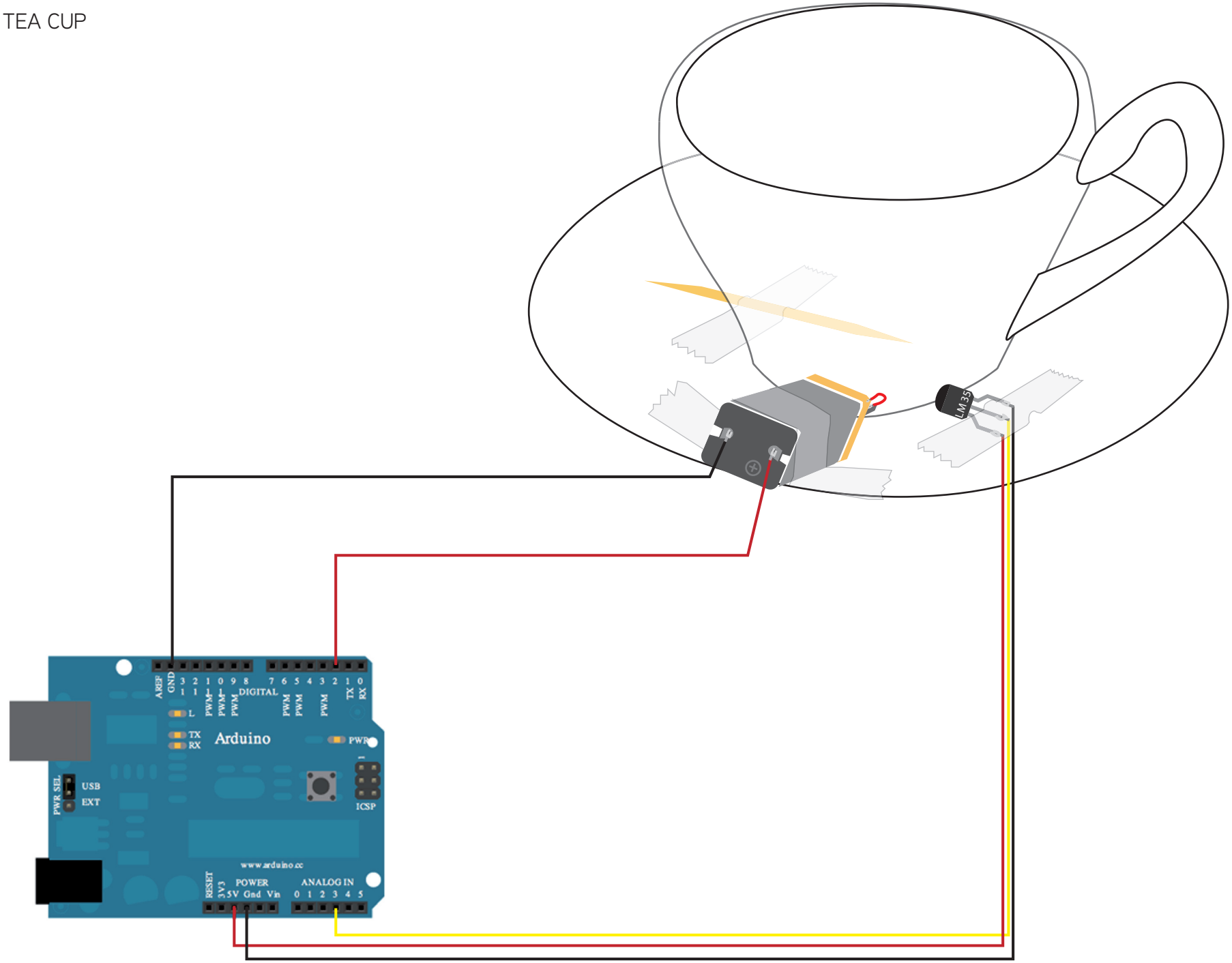
LM35 DZ temp sensor IC,0 to +100degC



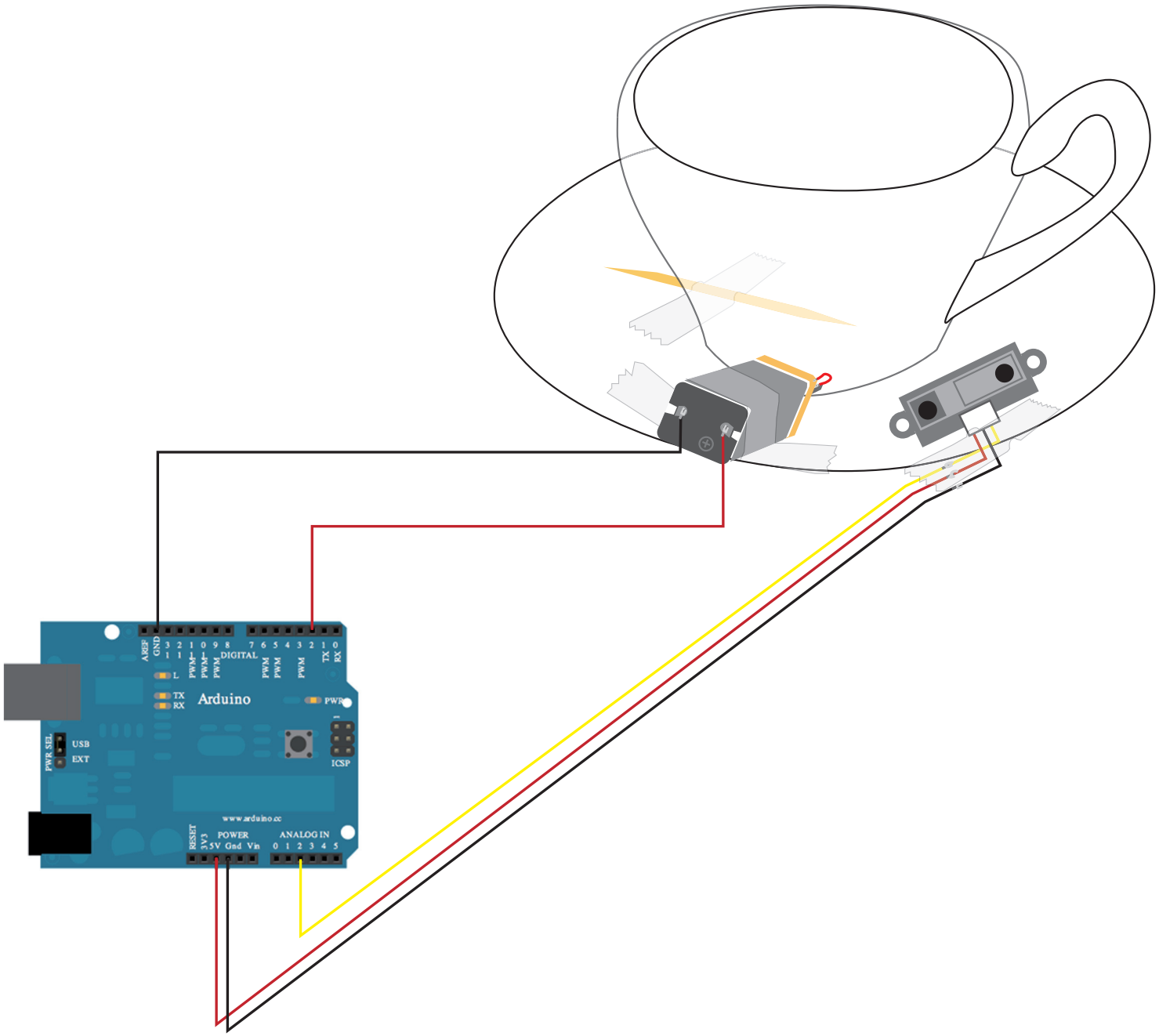
Distance Sensor 10-80cm,GP2Y0A21YK0F



FIXTURES TO TEA CUP



FIXTURES TO TEA CUP



SOME MORE CODING STUFF

constants

true/ false
OUTPUT/ INPUT
HIGH/ LOW

methods

`pinMode(pinNumber, mode)`	Configures the specified pin to behave either as an input or an output
`digitalWrite(value)`	Write a HIGH or a LOW value to a digital pin. If the pin has been configured as an OUTPUT with pinMode(), its voltage will be set to 5V
`int digitalRead(pinNumber)`	Reads the value from a specified digital pin, either HIGH or LOW.
`analogRead(pinNumber)`	Reads the value from the specified analog pin
`analogWrite(pin,value)`	Writes an analog value (PWM wave) to a pin. Can be used to light a LED at varying brightnesses or drive a motor at various speeds.
`delay(ms)`	Pauses the program for the amount of time (in miliseconds) specified as parameter.
`millis`	Returns the number of milliseconds since the Arduino board began running the current program.

WRITING A CODE:

what are our variables?

what do we want to meassure and where from?

when do we want the component to react and do what?

how often should that happen?

temp_sensor_motor_more_complex | Arduino 0022

File Edit Sketch Tools Help

temp_sensor_motor_more_complex §

```
const int motor_vibration_pin = 2;
const int temperature_pin = 3;

const int trigger_temp = 21;   // temperature at which we want to activate the vibrator
const int vibration_duration = 10; // seconds

boolean activated = false;

void setup()
{
    pinMode(motor_vibration_pin, OUTPUT);
    digitalWrite(motor_vibration_pin, LOW);
    Serial.begin(9600); // start serial communication
}

void loop()
{
    int current_temp = ( 5.0 * analogRead(temperature_pin) * 100.0) / 1024.0;

    Serial.print("Current temperature: ");

    Serial.print(current_temp,DEC);
    Serial.println(" Celsius");

    if (activated == false)
    {
        if (current_temp > trigger_temp + 5)
        {
            activated = true;
            Serial.println("5 Degrees above trigger temperature, starting to pay attention!");
        }
    }
    else
    {
        if (current_temp < trigger_temp)
        {
            Serial.println("Temperature falling below trigger temperature! Activate the MOTORRRRR!");

            digitalWrite(motor_vibration_pin, HIGH);
            delay(vibration_duration * 1000);       // in milliseconds, 10000ms = 10 seconds
            digitalWrite(motor_vibration_pin, LOW);
            activated = false;
            Serial.println("Done. Waiting for higher temperature again.");
        }
    }
    delay(1000);
}
```

ARDUINO CODE
- temperature sensor

ARDUINO CODE
- distance sensor

distance_sensor_motor | Arduino 0022

File Edit Sketch Tools Help

distance_sensor_motor

```
// Distance triggered saucer vibrator

const int motor_vibration_pin = 2;
const int distance_pin = 3;

const int trigger_dist = 400;   // distance at which we want to activate the vibrator

void setup()
{
  pinMode(motor_vibration_pin, OUTPUT);
  digitalWrite(motor_vibration_pin, LOW);
  Serial.begin(9600); // start serial communication

}

void loop()
{
  int current_dist = analogRead(distance_pin);
  Serial.print("Current dist: ");
  Serial.println(current_dist);

  if (current_dist > trigger_dist)
  {
    Serial.println("Distance falling below trigger distance! Activate the MOTORRRRR!");
    digitalWrite(motor_vibration_pin, HIGH);
  }
  else
  {
    Serial.println("Distance above trigger distance! Switching off the MOTORRRRR!");
    digitalWrite(motor_vibration_pin, LOW);
  }
  delay(100);
}
```

1

sources & further resources:

- ‘Programming Interactivity - A Designer’s Guide to Processing, Arduino and openFrameworks’ by Joshua Noble
- ‘Making Things Talk: Practical Methods for Connecting Physical Objects’ by Tom Igoe
- http://www.arduino.cc/

katrin.baumgarten@network.rca.ac.uk