Arduino Fever - PHYSICAL COMPUTING

Wolf Paulus . com

Arduino
955 - 1015

Taking advantage of strong disagreements inside Germany, the Italian feudatories rebelled and, in 1002, elected Arduino, Marquess of Ivrea, king of Italy.

He distinguished himself particularly because of his fights against the Bishop of Ivrea.

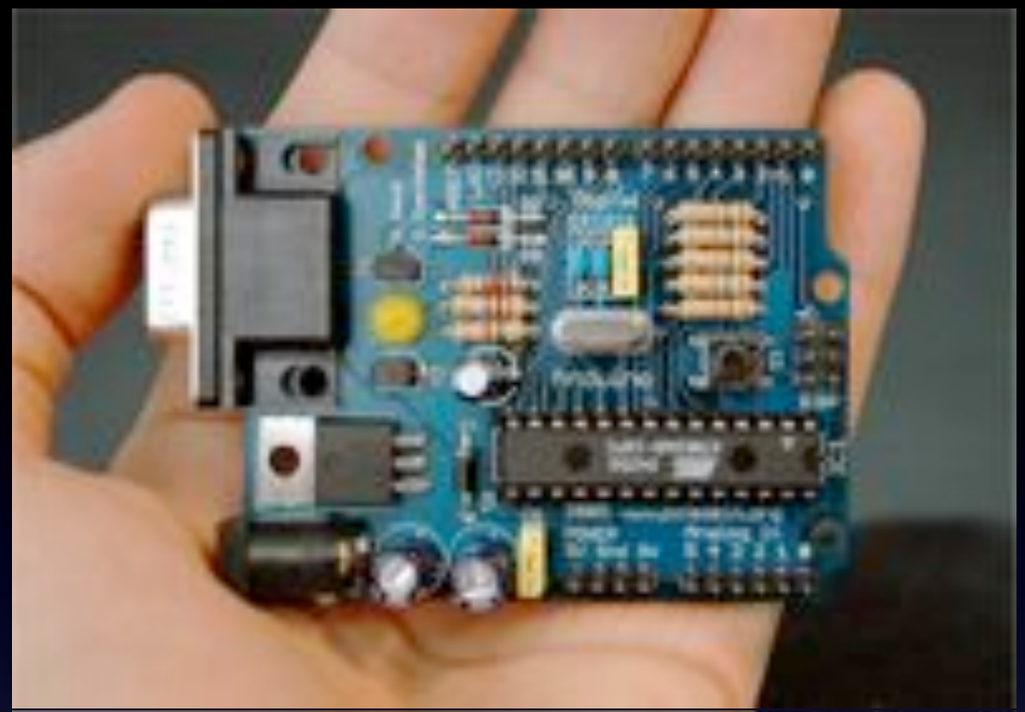
Arduino Fever

disruptive innovation

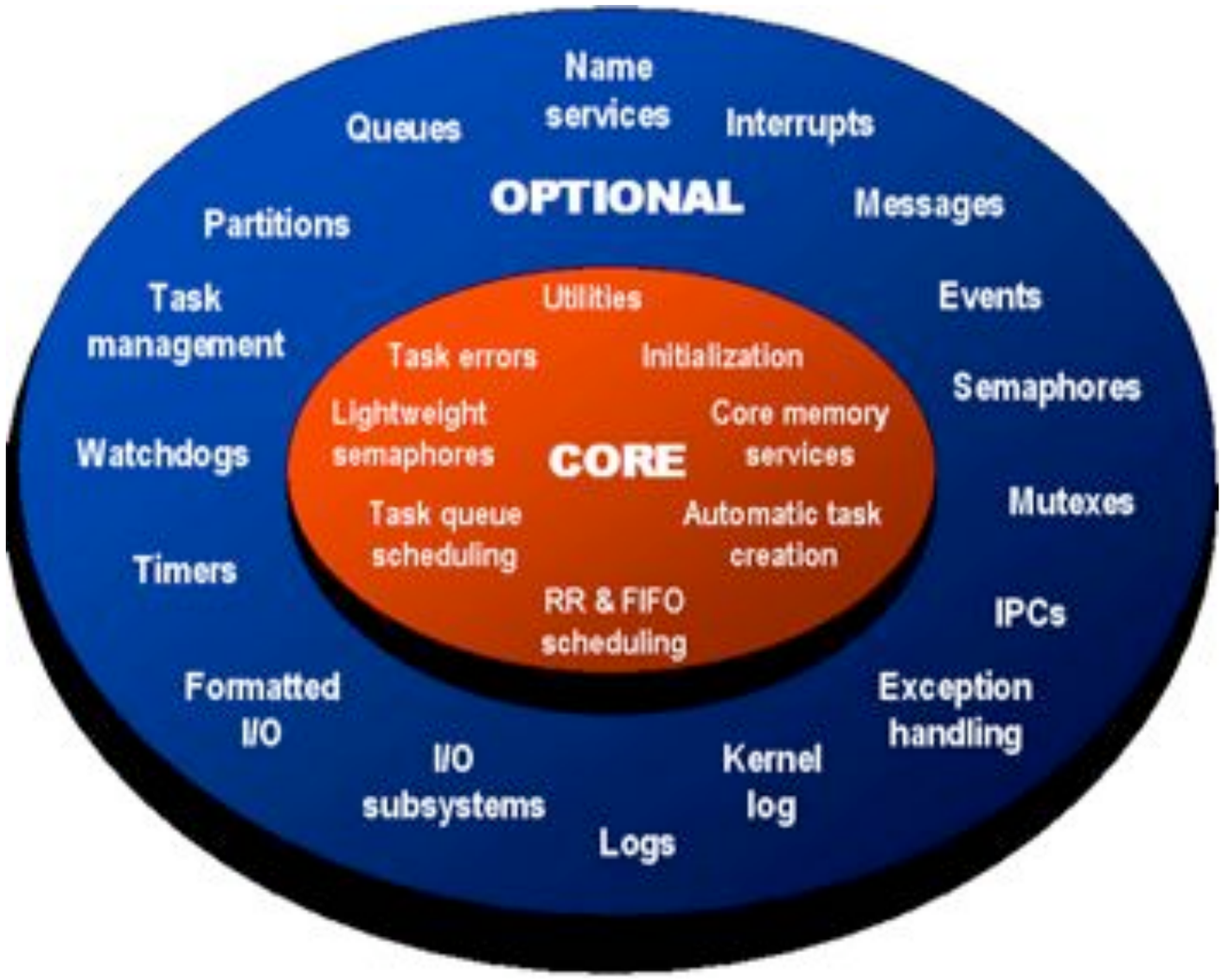
SoCal Code Camp Jan. 27, 2007

disruptive innovation

SoCal Code Camp Jan. 27, 2007



Arduino is an open-source computing platform based on a simple board, and a development environment for writing software.


Name services
Queues
Interrupts
OPTIONAL
Partitions
Messages
Task management
Utilities
Events
Task errors
Initialization
Semaphores
Lightweight semaphores
Core memory services
Watchdogs
CORE
Mutexes
Task queue scheduling
Automatic task creation
Timers
RR & FIFO scheduling
IPCs
Formatted I/O
Exception handling
I/O subsystems
Kernel log
Logs

The 1st Arduino was produced in Jan. 2005 by **David Cuartielles** and **Massimo Banzi,** teachers of Physical Computing and Interaction Design at the Interaction Design Institute, Milan Italy.

- **Interaction Design**
  design of any interactive experience

- One area of Interaction Design is **Physical Computing**, dealing with electronic prototyping and turning micro-controllers, sensors, actuators etc. into materials used by artists and designers.

“**Physical Computing** is an approach to learning how humans communicate through computers that starts by considering how humans express themselves physically”

Physical Computing

Physical Computing

SoCal Code Camp Jan. 27, 2007

cheap, about US$ 30

based on a simple I/O board

open-source physical computing platform

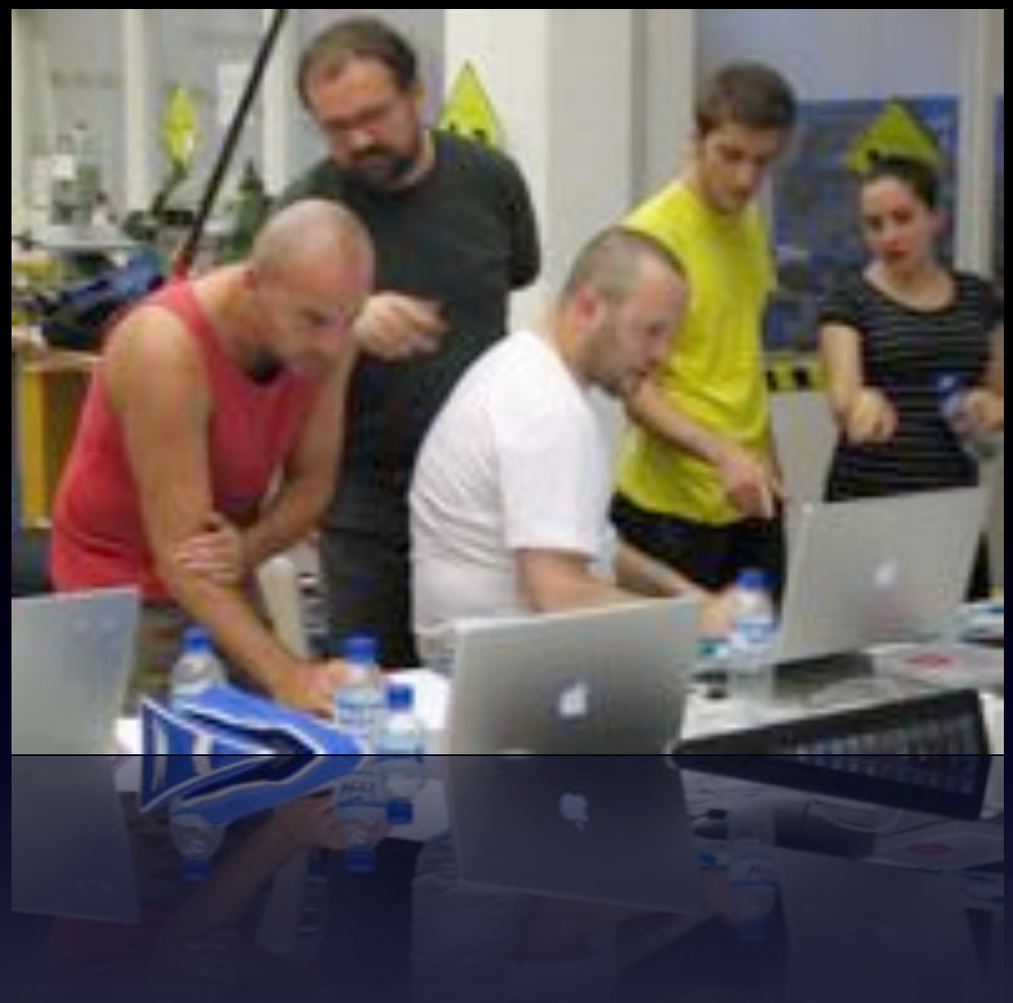
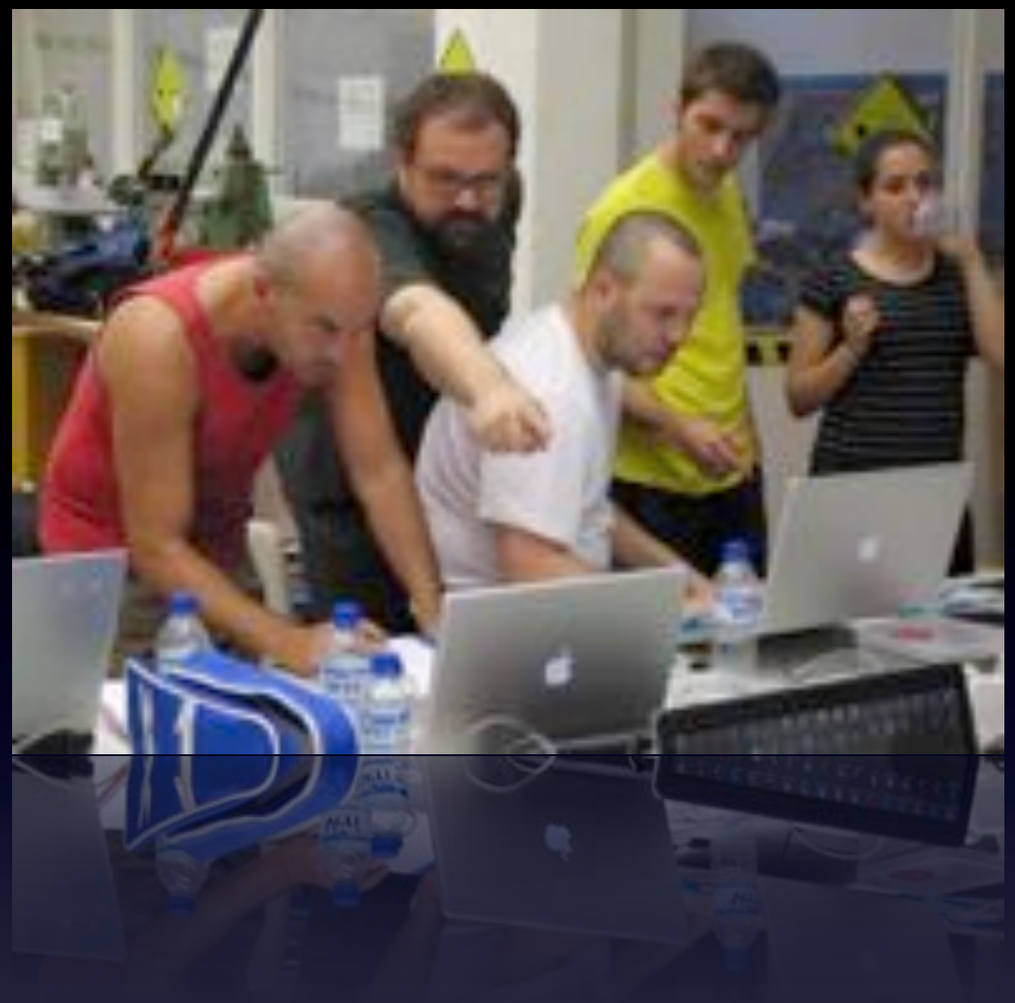
standalone interactive object or connected to a computer

Arduino Boards can be assembled or purchased pre-assembled

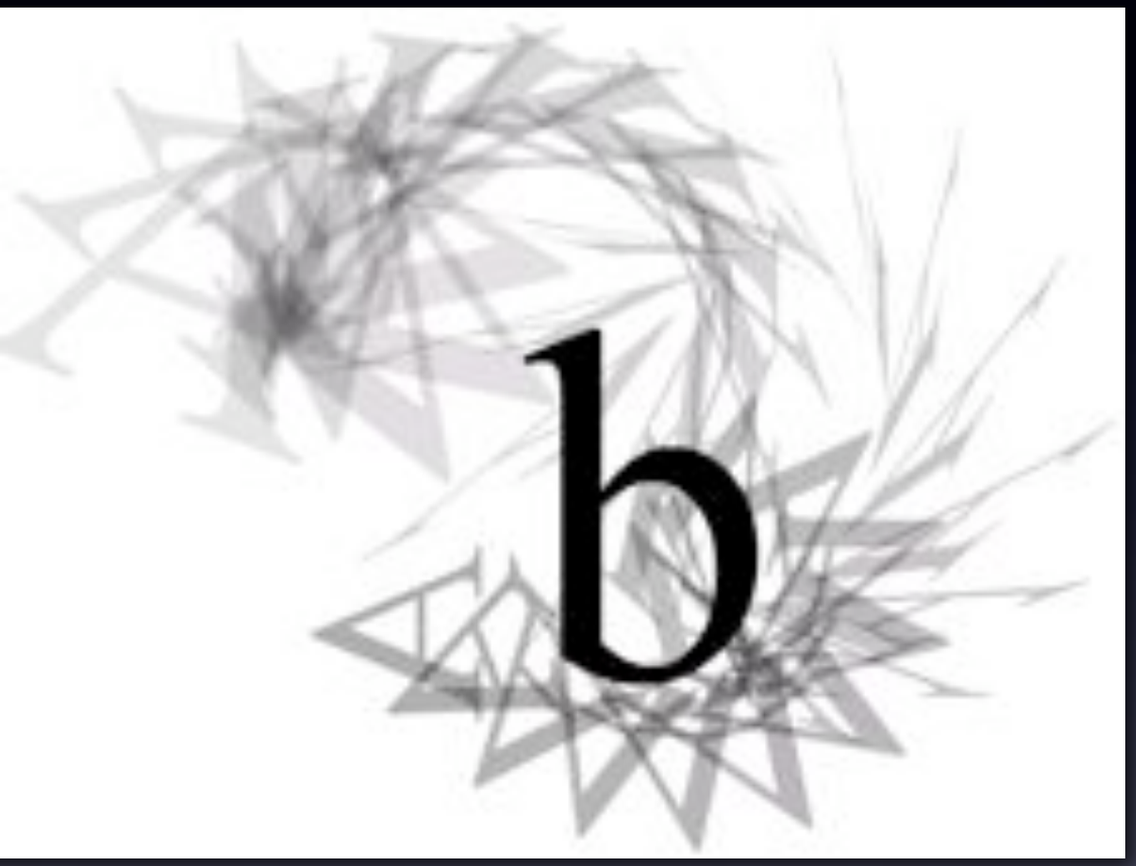

Arduino – 0006 Alpha
led_blink §
int p = 13; // LED connected to digital pin 13
void setup()
{
pinMode(p, OUTPUT); // sets the digital pin as output
}
void loop()
{
digitalWrite(p, HIGH); // sets the LED on
delay(1000); // waits for a second
digitalWrite(p, LOW); // sets the LED off
delay(1000); // waits for a second
}
Done compiling.
Binary sketch size: 4300 bytes (of a 7168 byte maximum)

- May 2005: Wiring 1.0 (ALPHA) is released.
- Aug 2005: Arduino boards are assembled in Europe
- Aug 2005: *Tom Igoe* (NYU) assembles a small number of Arduino boards in the US.
- Oct 2005: Board are printed in batches of 1000
- Mar 2006: Arduino feat. in O'Reilly Make Magazine
- Nov 2006: Arduino feat. at ARTECH 2006

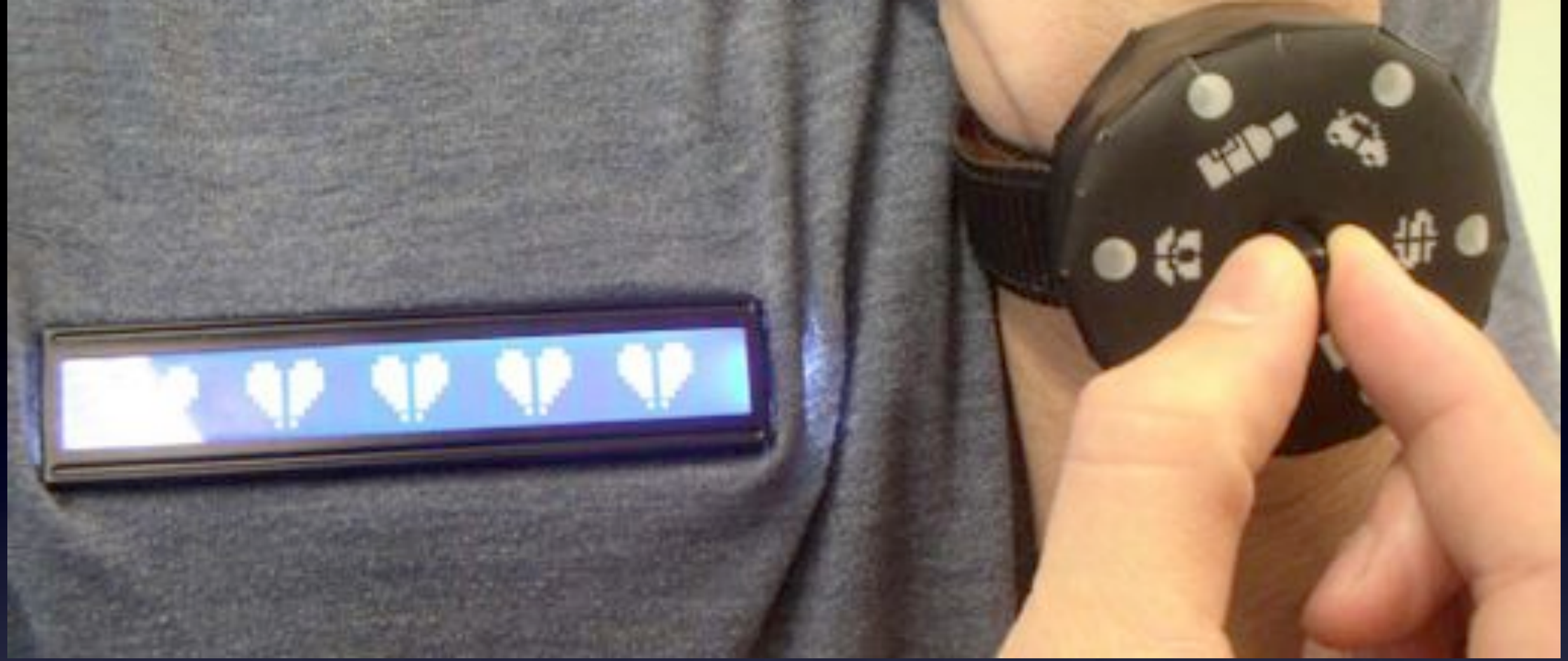
Tom Igoe, ITP NYU
assembles Arduino Boards, August 2005

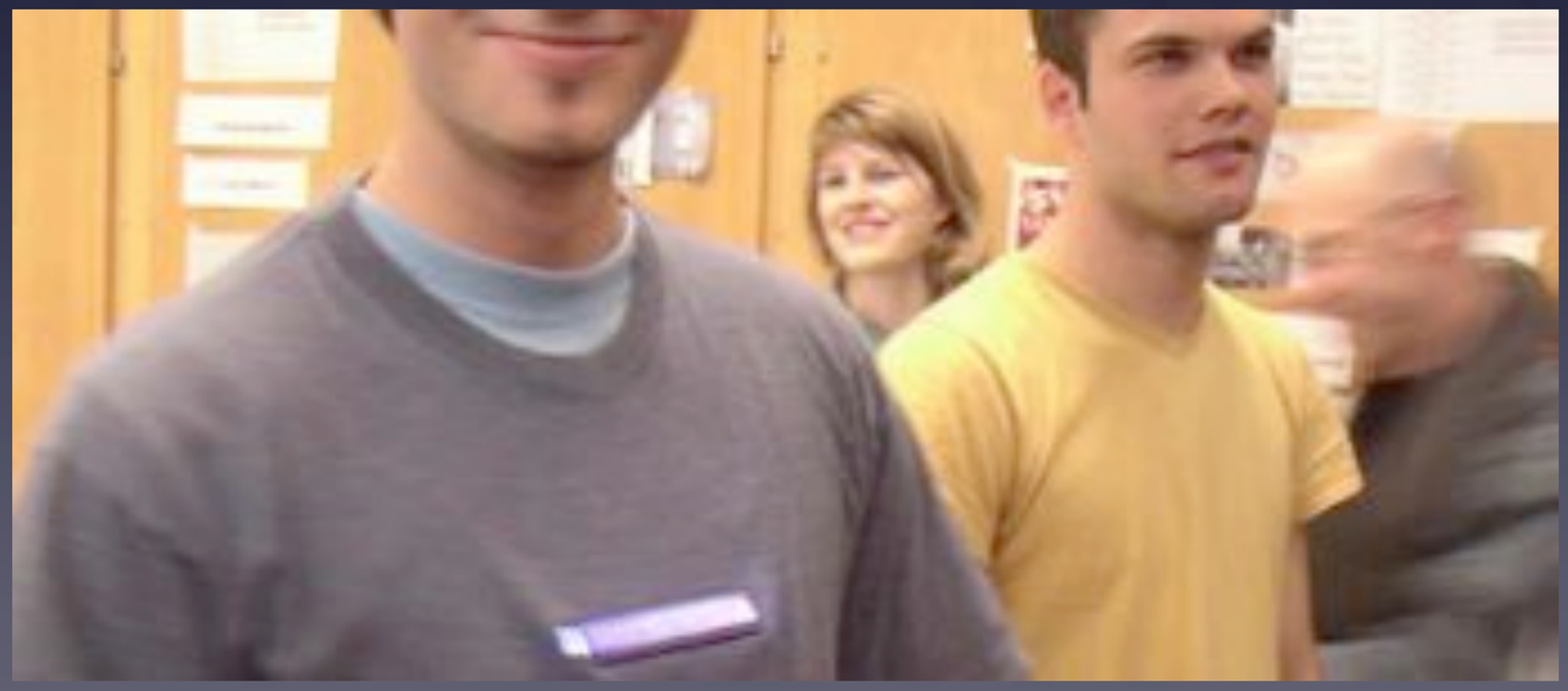



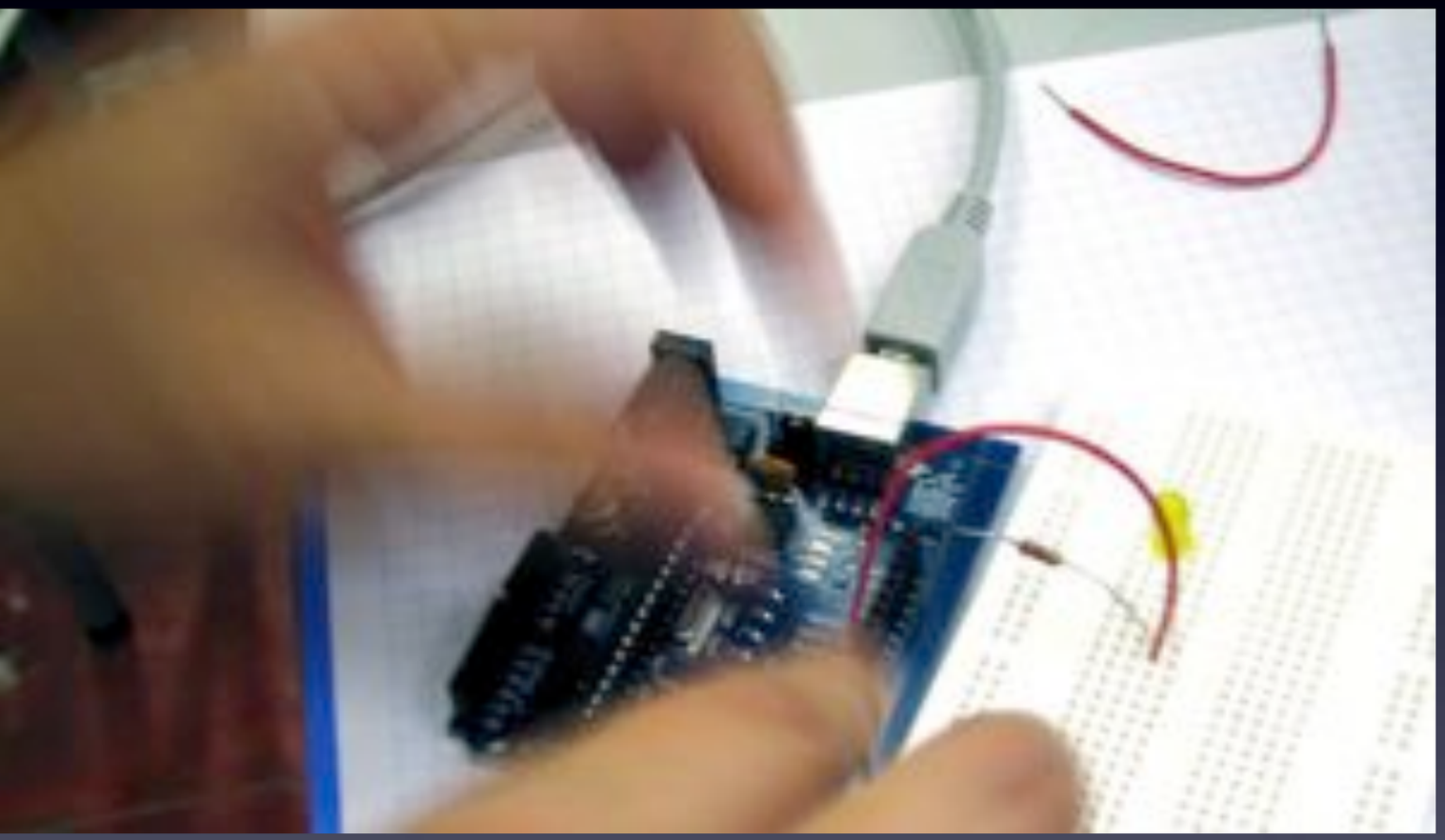
- really easy to program
- Java-like language
- allows quickly creation of prototypes
- cheap hardware makes mistakes tolerable



Classic engineering relies on a strict process for getting from **A** to **B**, while the *arduino way* is based on maybe getting lost in the way and finding **C** instead ... *"tinkering"*







- circuit board
- IDE
  Integrated Development Environment
- sample code


Digital
arduino
www.arduino.cc
Analog in

16 MHz-Crystal

Power-LED

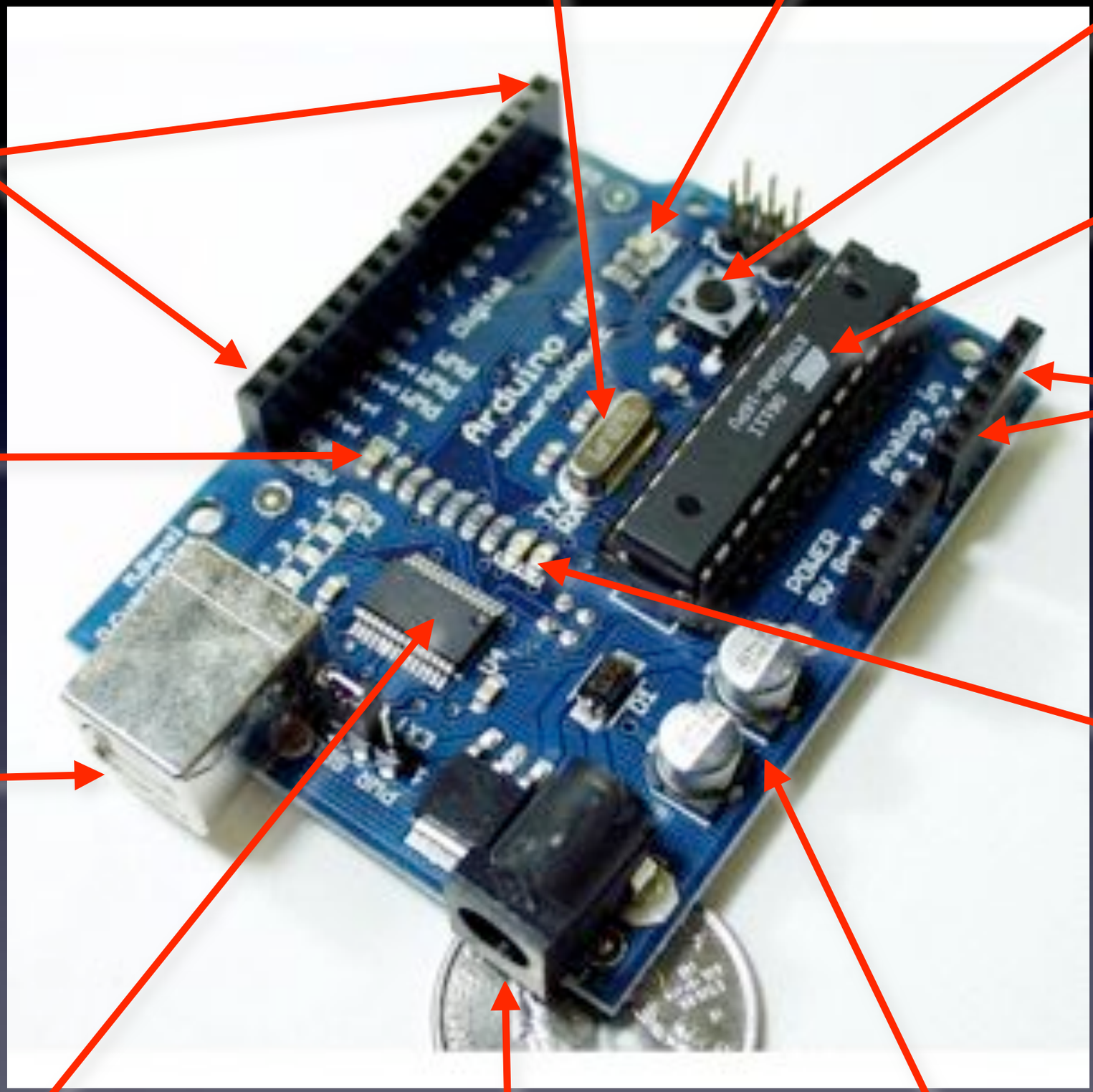
Reset-Button

digital I/O

ATmega8
Micro

analog
input

Test LED



USB

TX/RX
LEDs

FTDI USB

Power

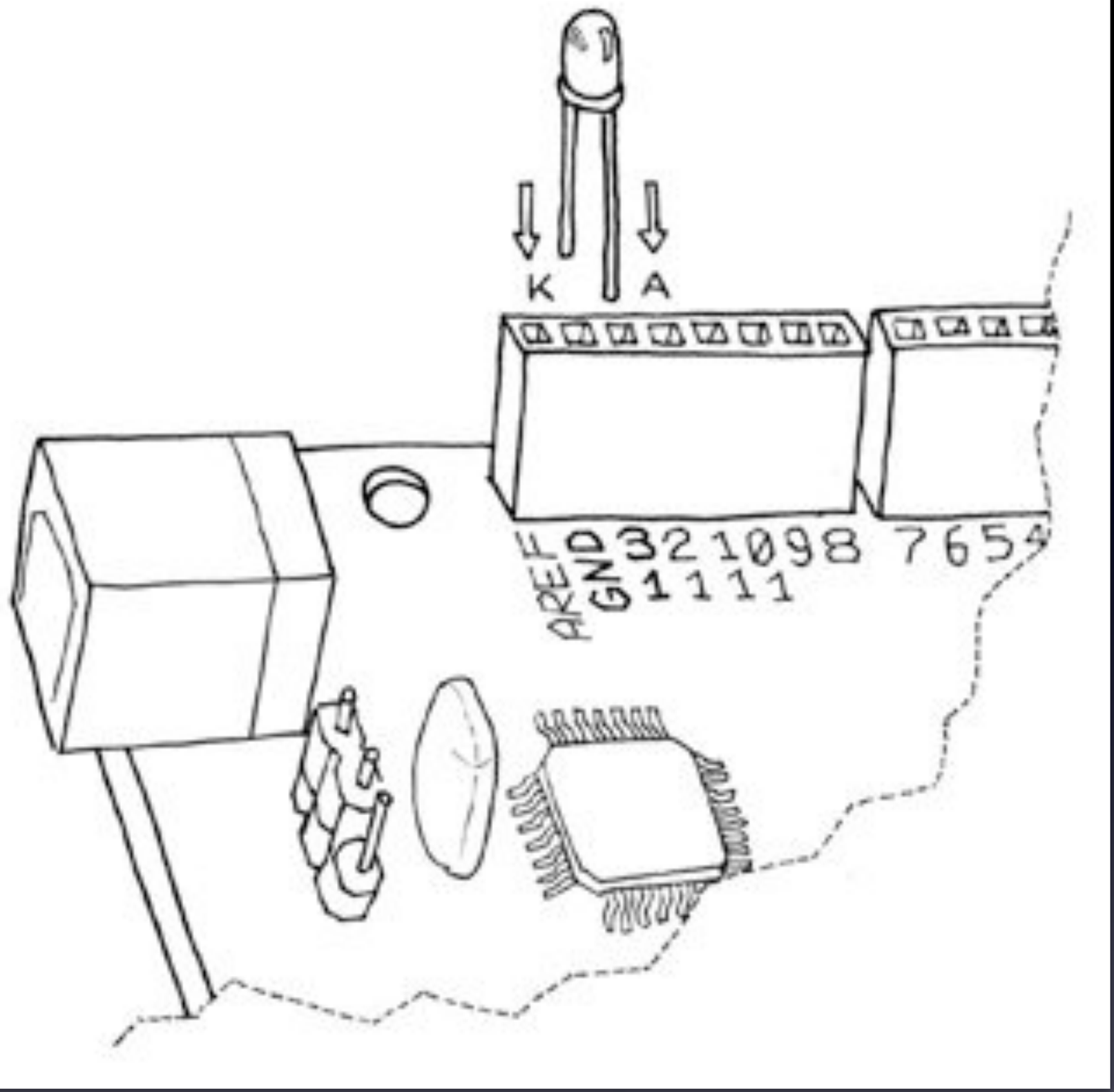
Power Regulator

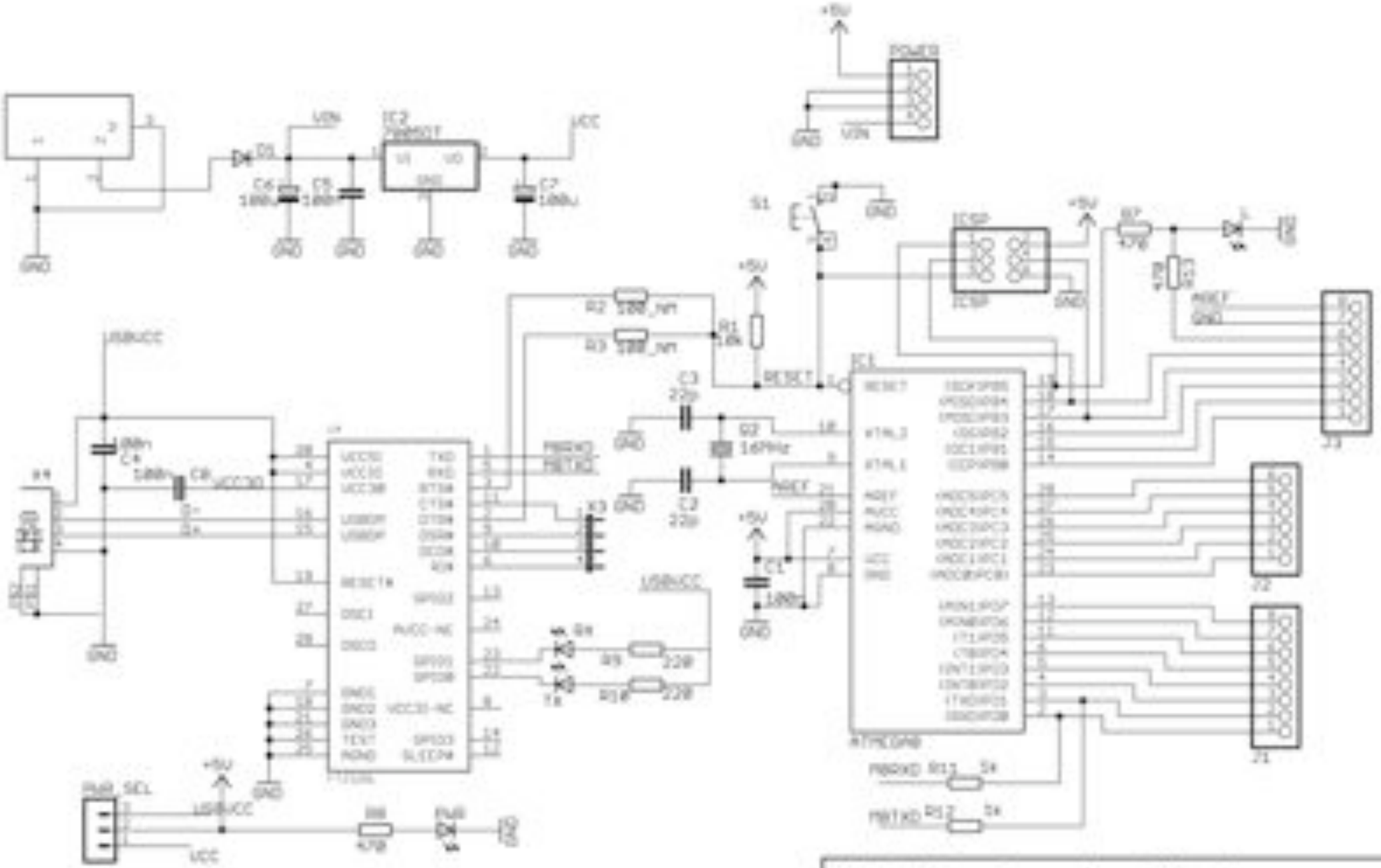
- 16 MHz
- 8 KByte Flash RAM
  (1K taken by the boot loader)
- 1 KByte RAM
  e.g. for auto/local variables and stack
- 14 digital Input/Output Ports

ATmega8

- USB 2.0 compatible
- Transmit and receive LED drive signals
- 256 Byte receive, 128 Byte transmit buffer
- Data transfer rate from 300 bit/s to 3 Mb/s

Single chip USB to async. serial data transfer interface



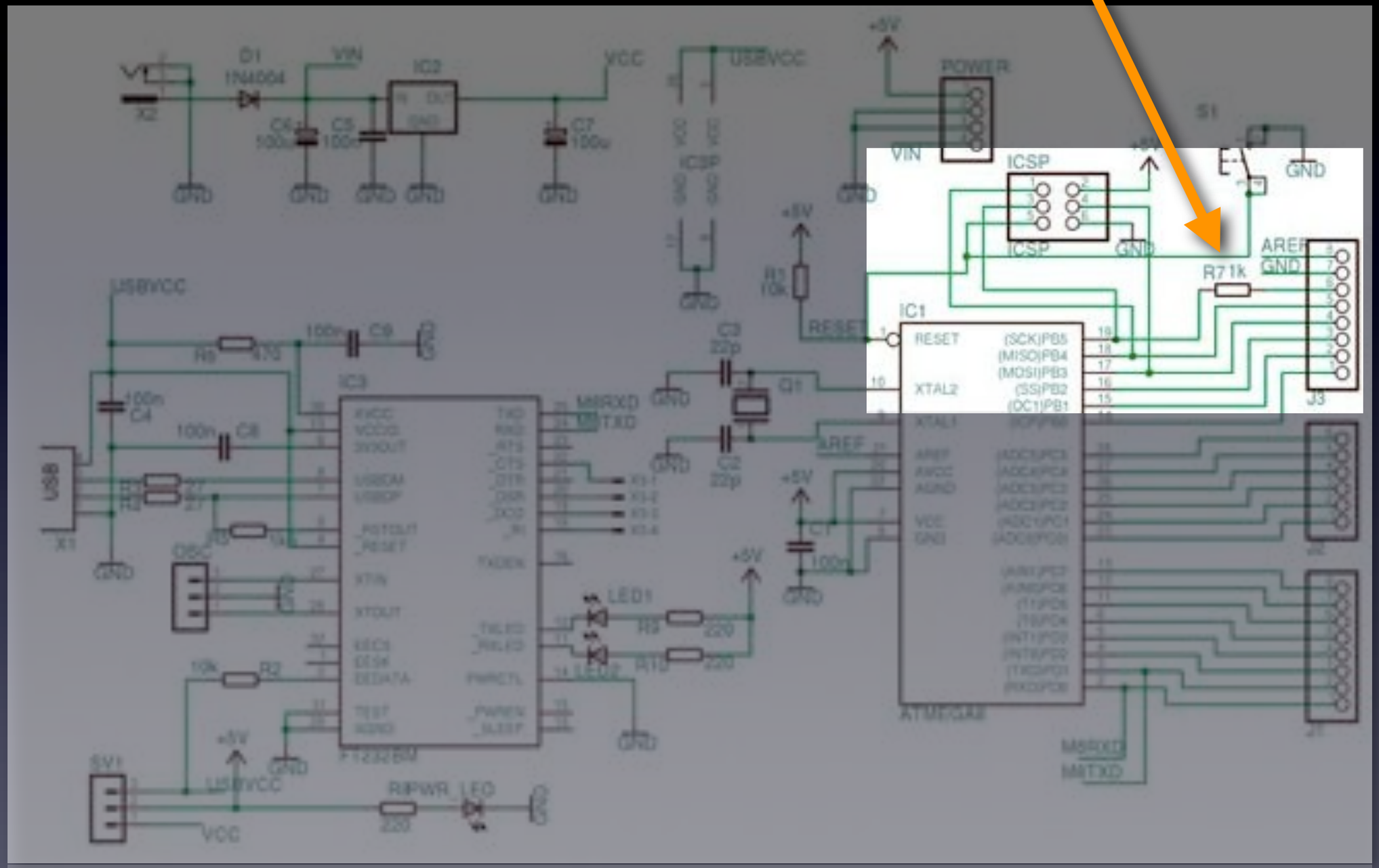


Arduino Nuova Generazione (NG) v.4.0
Part of the Arduino project http://www.arduino.cc

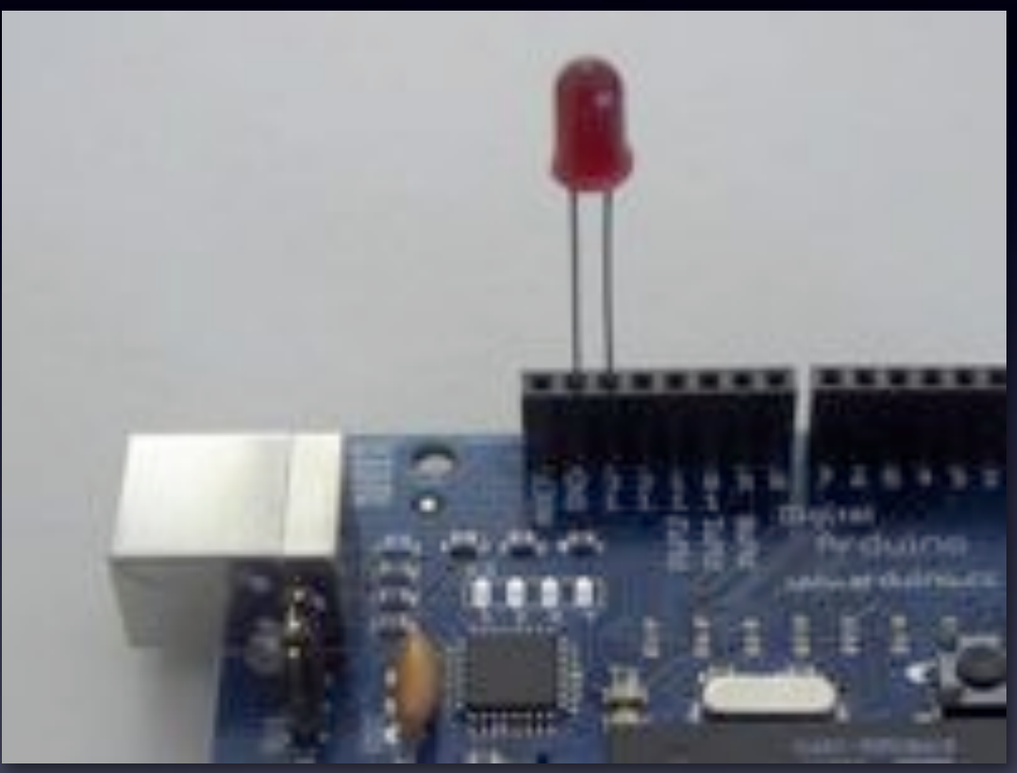
Designed in Italy by the Arduino Team
Engineered by Gianluca Martino http://www.smartprojects.it

Made in Italy

1KOhm Resistor



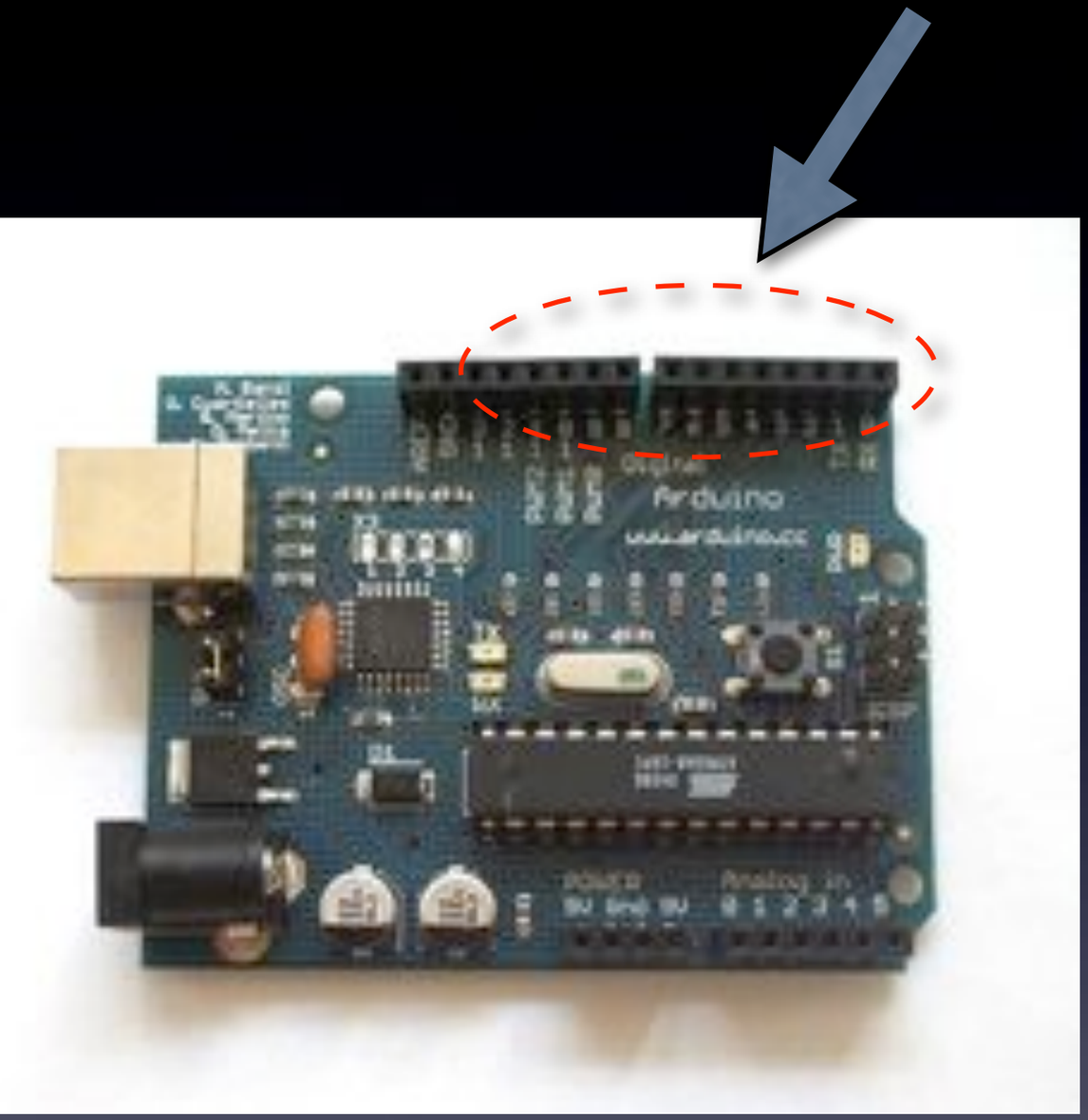
"hello world" - the blinking LED



```
int ledPin = 13;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);
  delay(1000);
  digitalWrite(ledPin, LOW);
  delay(1000);
}
```

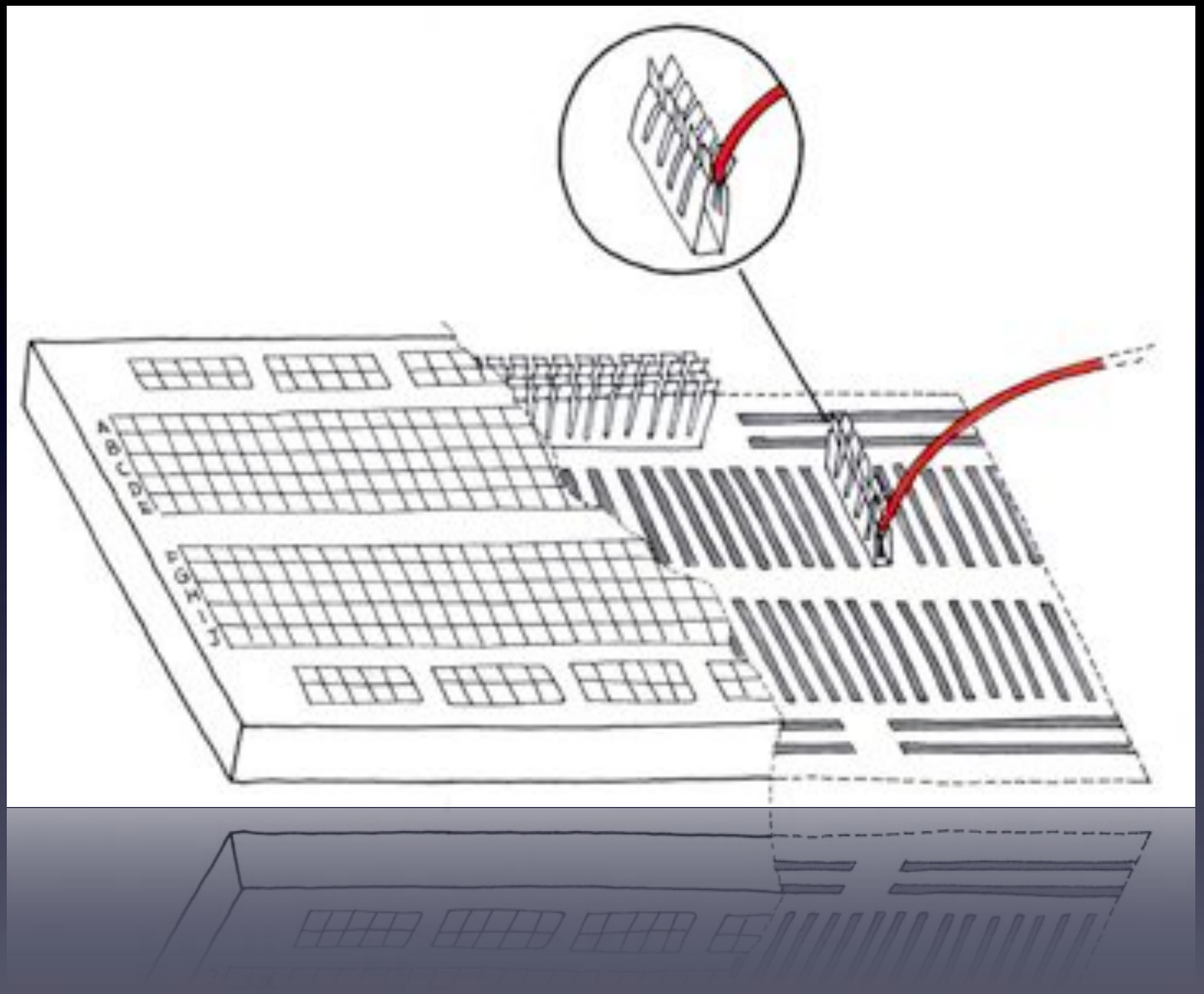
"hello world"

14 digital IO (pins 0..13) can be set as input or output ports.

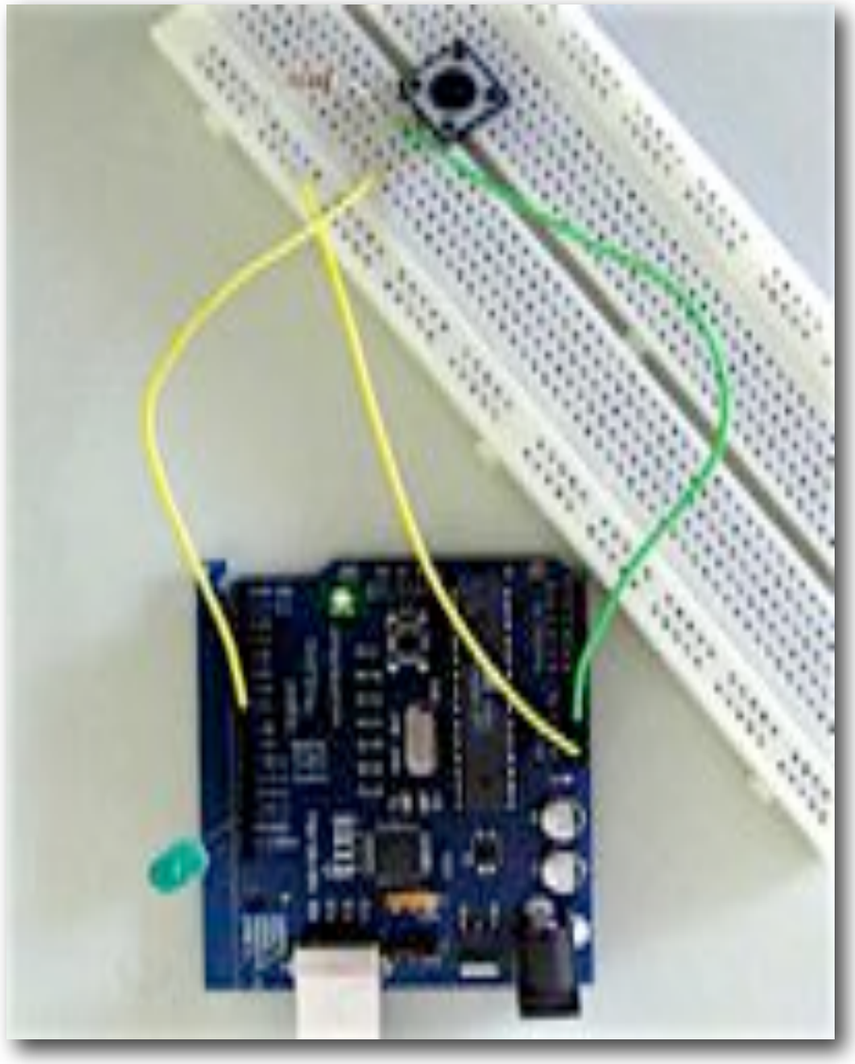


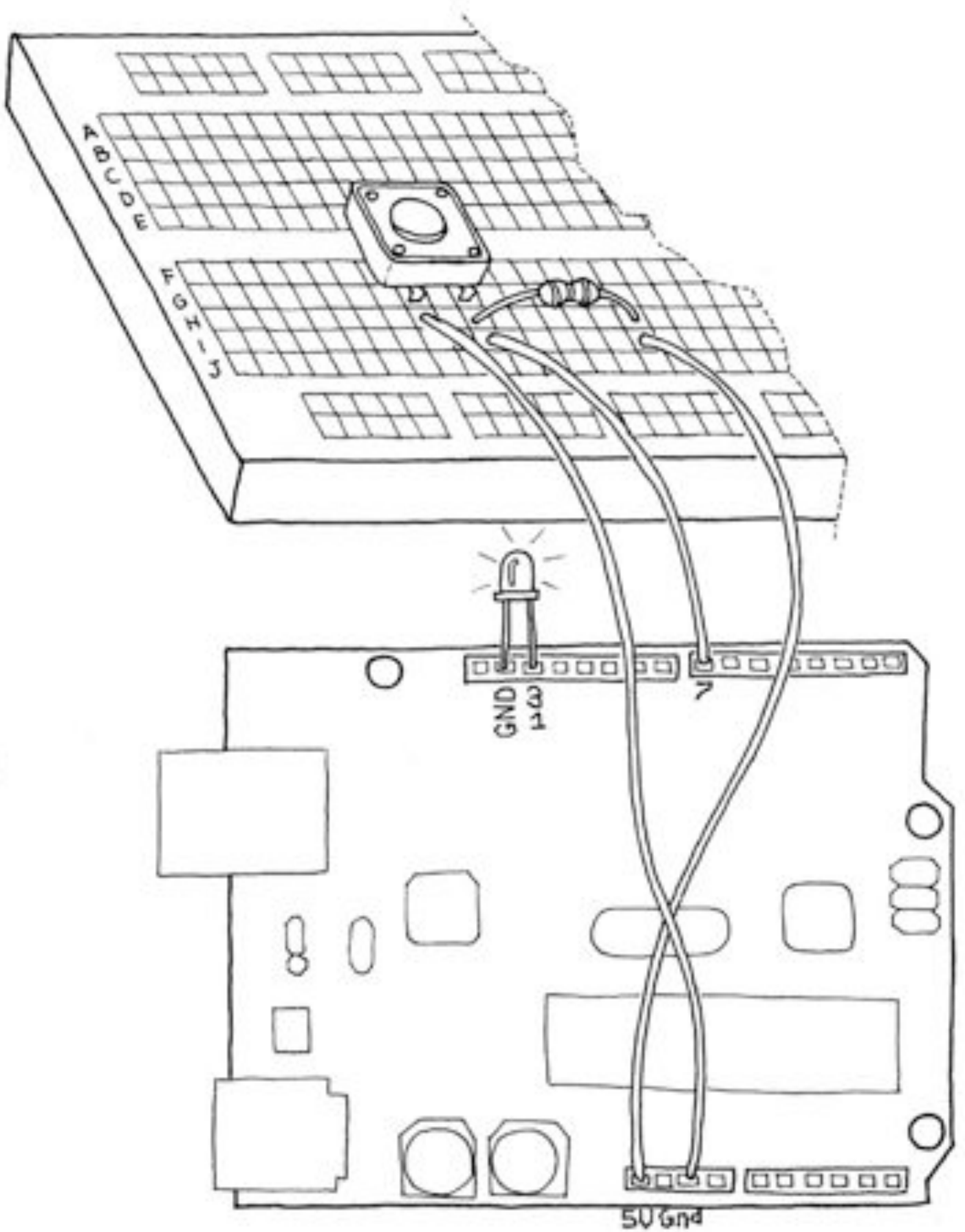
```
int ledPin = 13;

void setup() {

  pinMode(ledPin, OUTPUT);

}
```

Digital I/O Ports

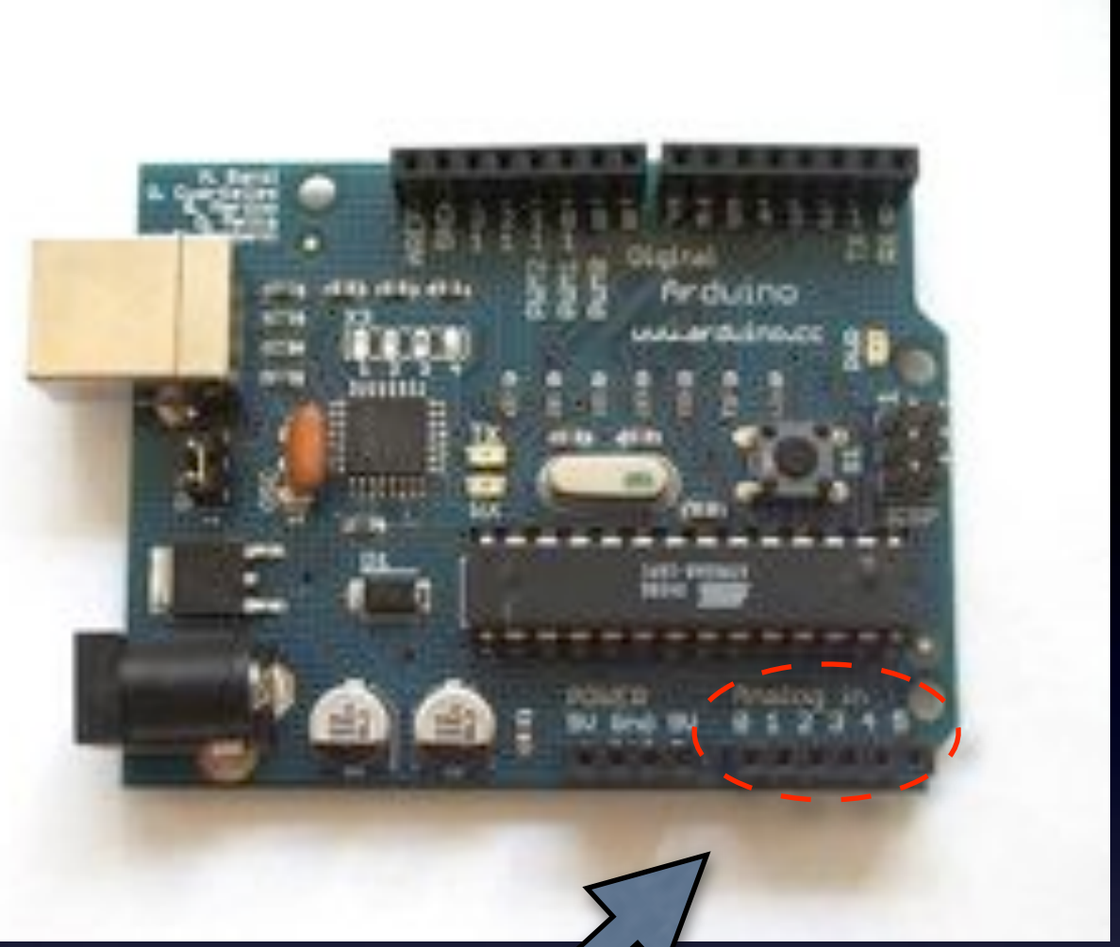


Solderless Breadboards





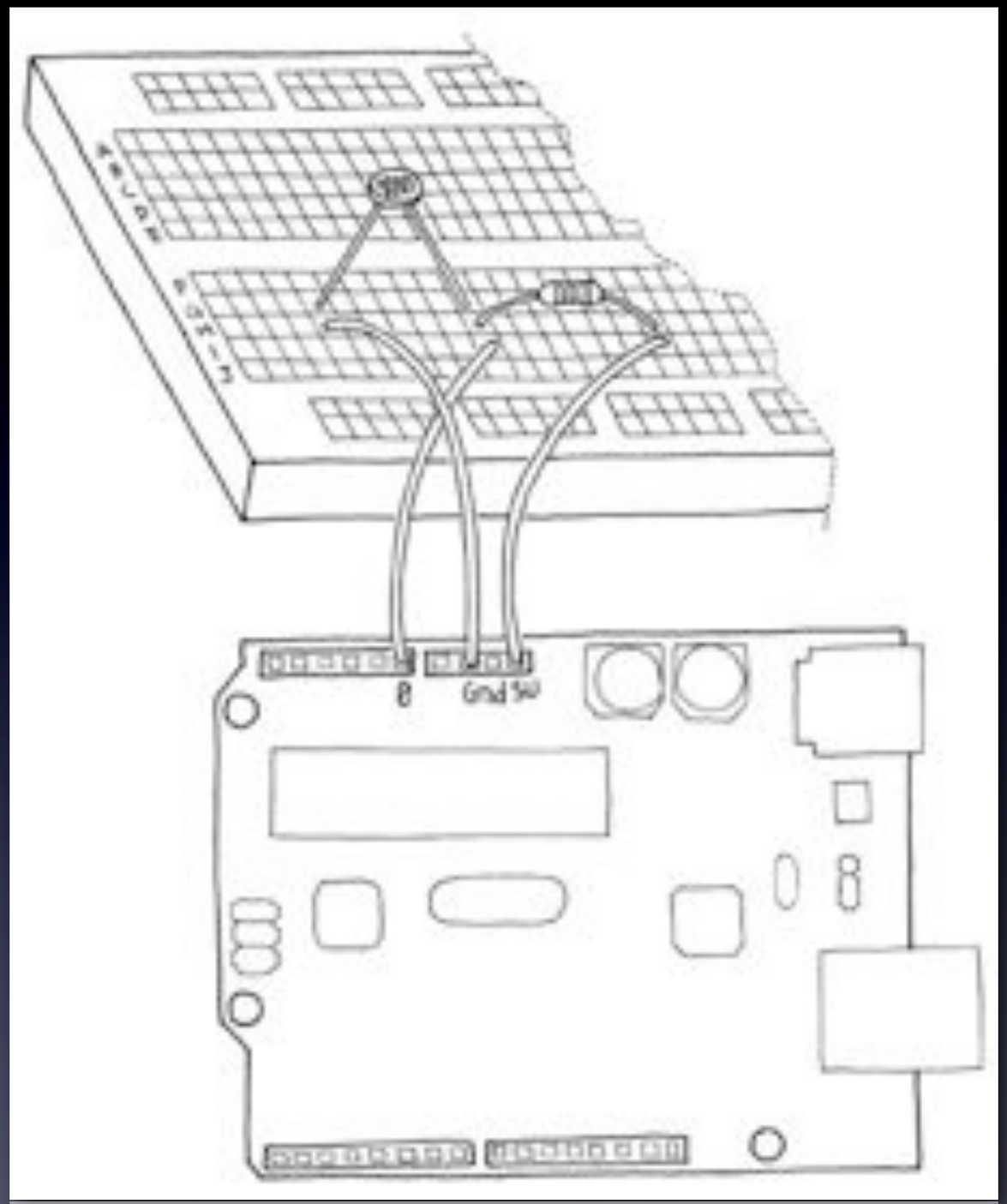
Digital I/O Ports




```
int analogPin = 3;
int val = 0;

void setup() {
   ..
}

void loop() {
   val=analogRead(analogPin);
   ..
}
```

6 analog input (pins 0..5), which take analog values (like voltage readings) and convert them into integer values 0..1023

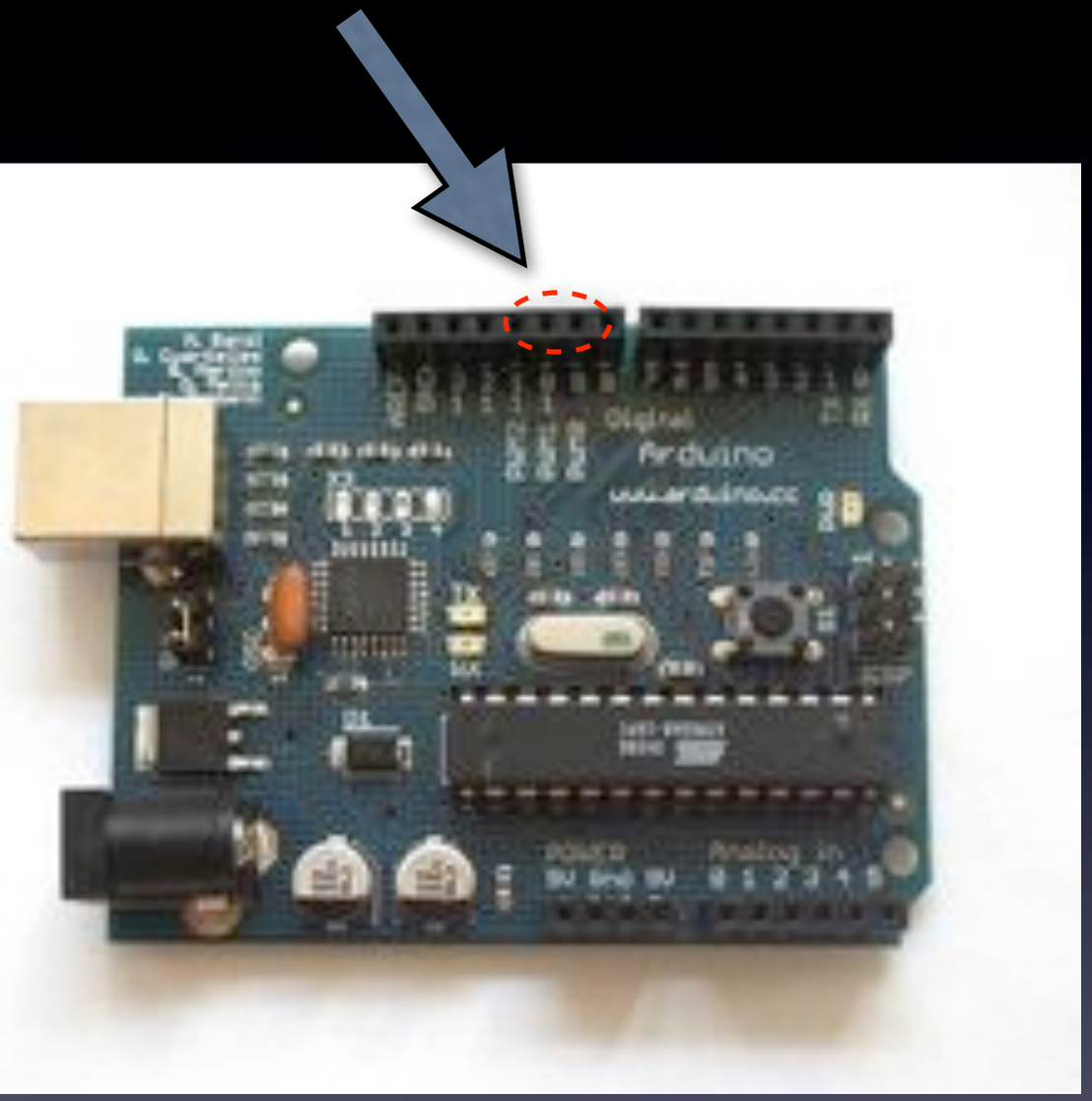
Analog In



Analog Input

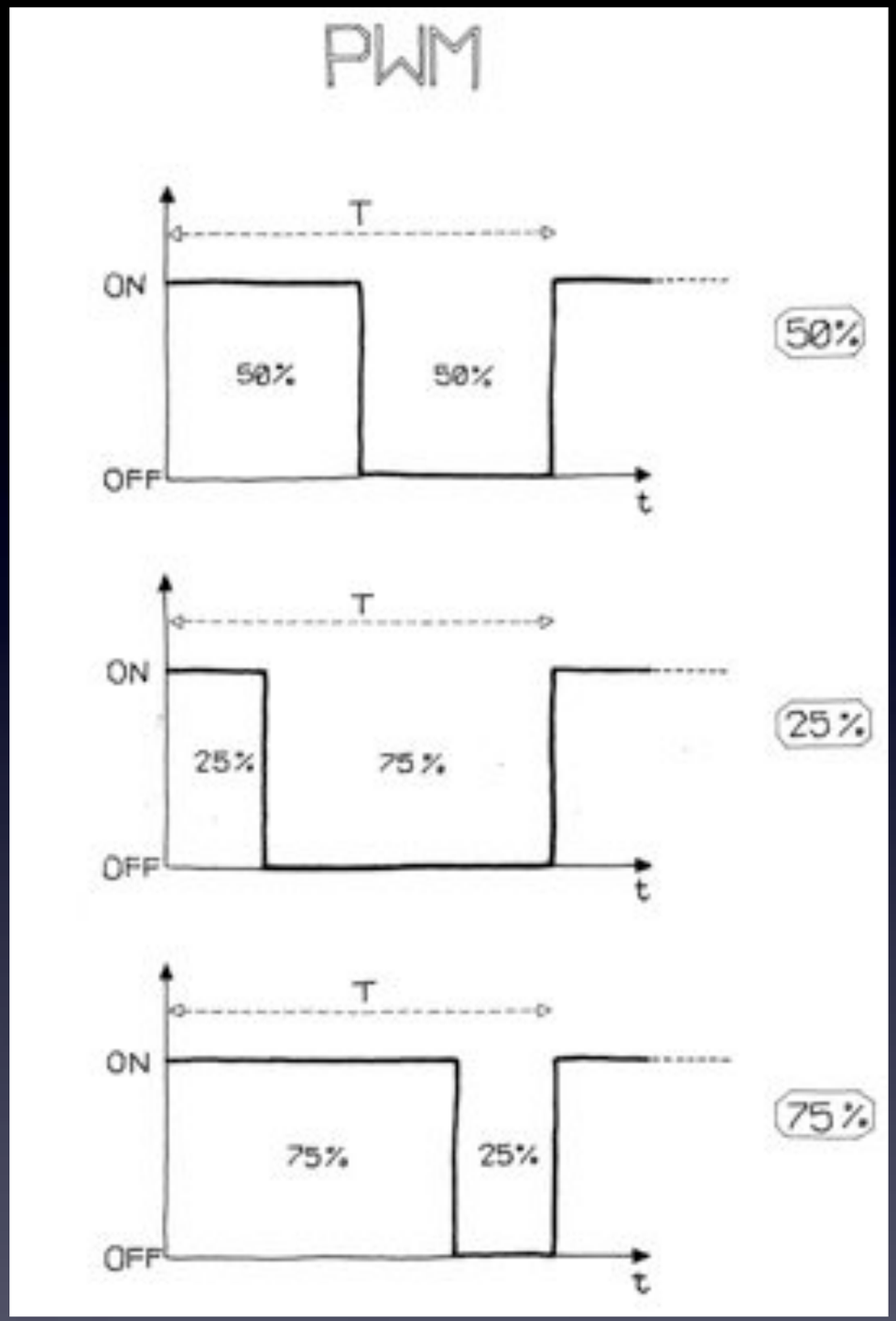
SoCal Code Camp Jan. 27, 2007

3 analog output (pins 9,10,11), reassigned digital pins take analog values 0..255 (0V..5V)

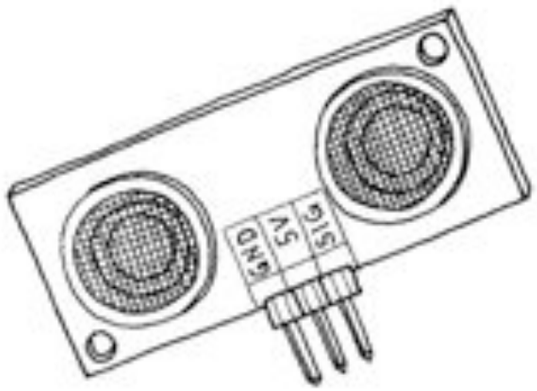


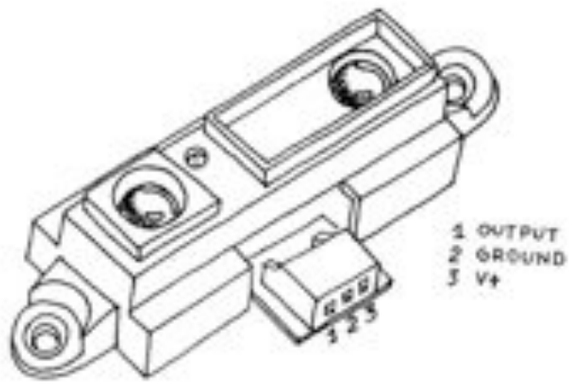

```
int analogPin = 3;
int ledPin = 9;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  val=analogRead(analogPin);
  analogWrite(ledPin, val/4);
}
```

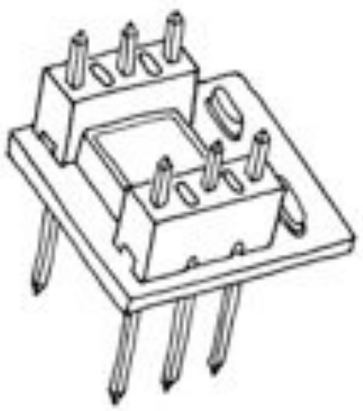
Analog Out

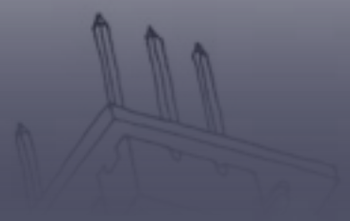


Analog Input









Sensors

SoCal Code Camp Jan. 27, 2007

- Temperature Probe
- Digital Themometer
- Temperature/Humidity Sensor

Temperature Sensors

IR Sensor detects changing patterns of passive infrared to sense motion from up to 20 feet away!

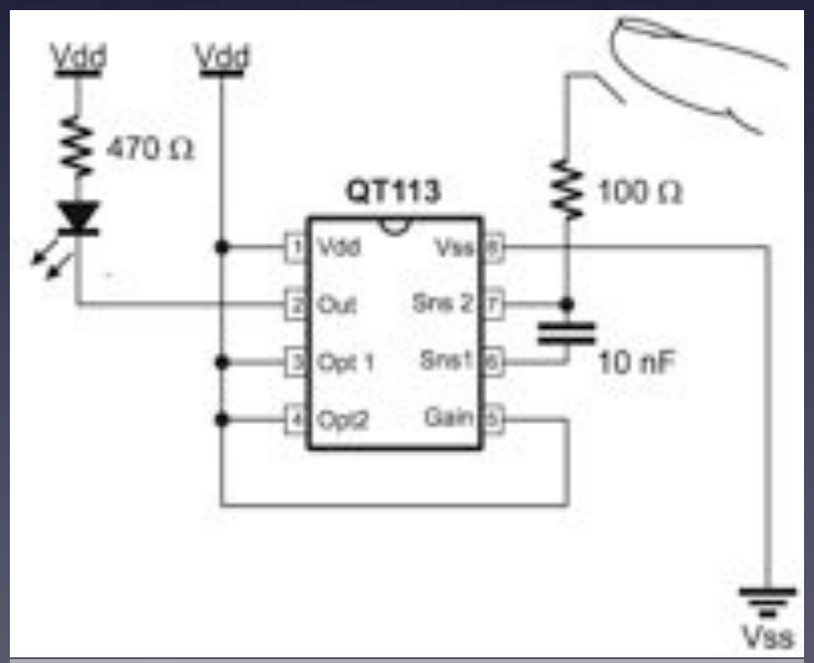
Passive Infra-Red Sensor

3-Axis Accelerometer can sense gravitational (g) force of ±3g on three axes (X, Y, and Z).

Accelerometer

SoCal Code Camp Jan. 27, 2007

- Color Sensor
- Light to Frequency Converter
- Touch Sensor



Sensors

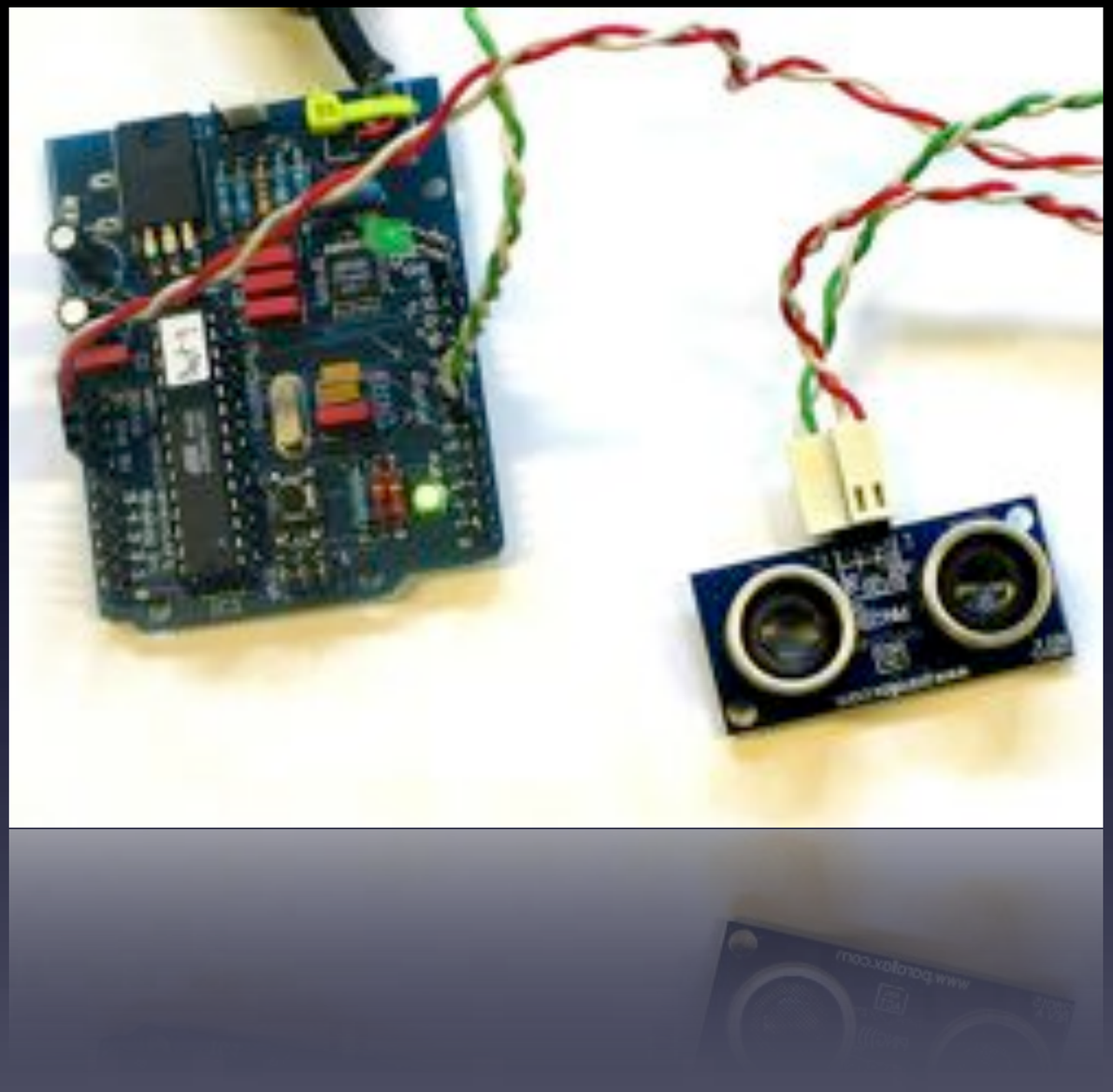
SoCal Code Camp Jan. 27, 2007

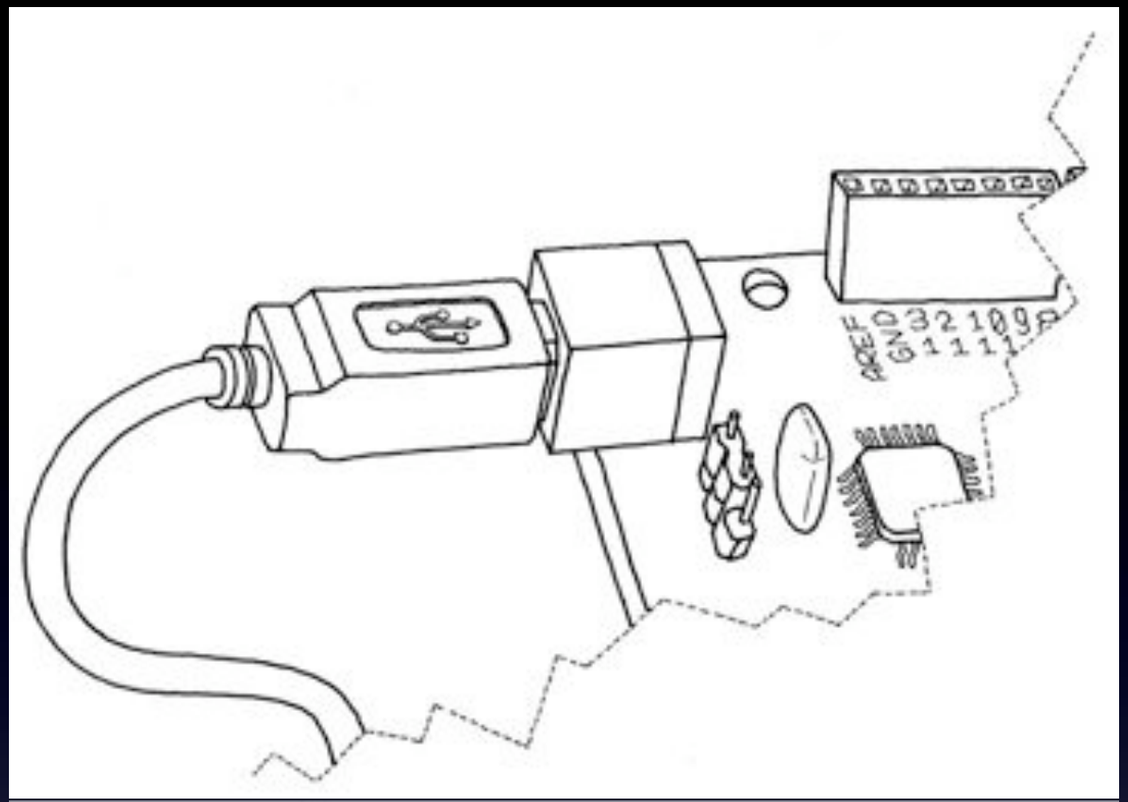
The module provides current time, date, latitude, longitude, altitude, speed and travel direction/ heading

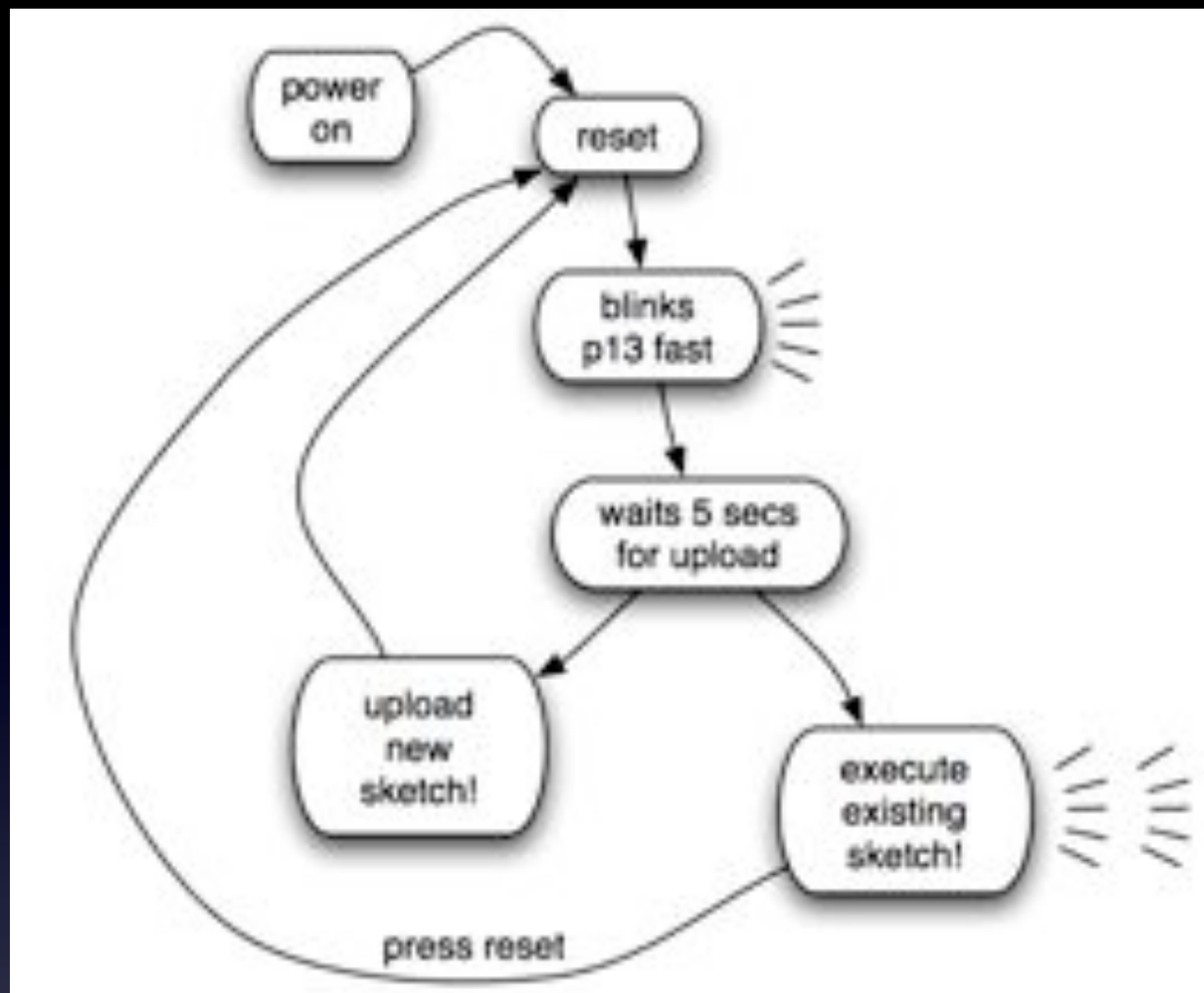
GPS Receiver Module

SoCal Code Camp Jan. 27, 2007

Ultrasonic Sensor



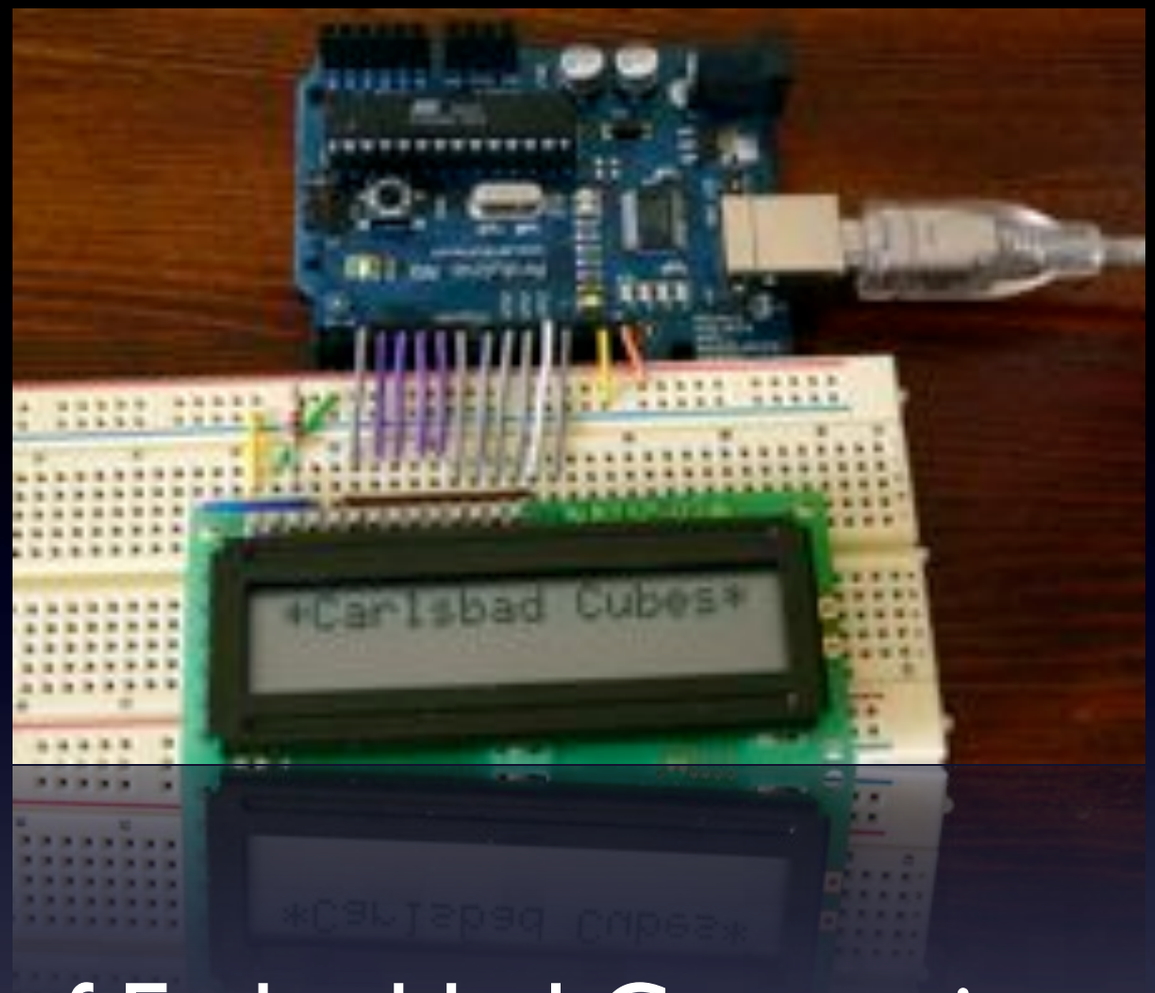




Arduino Board Life Cycle

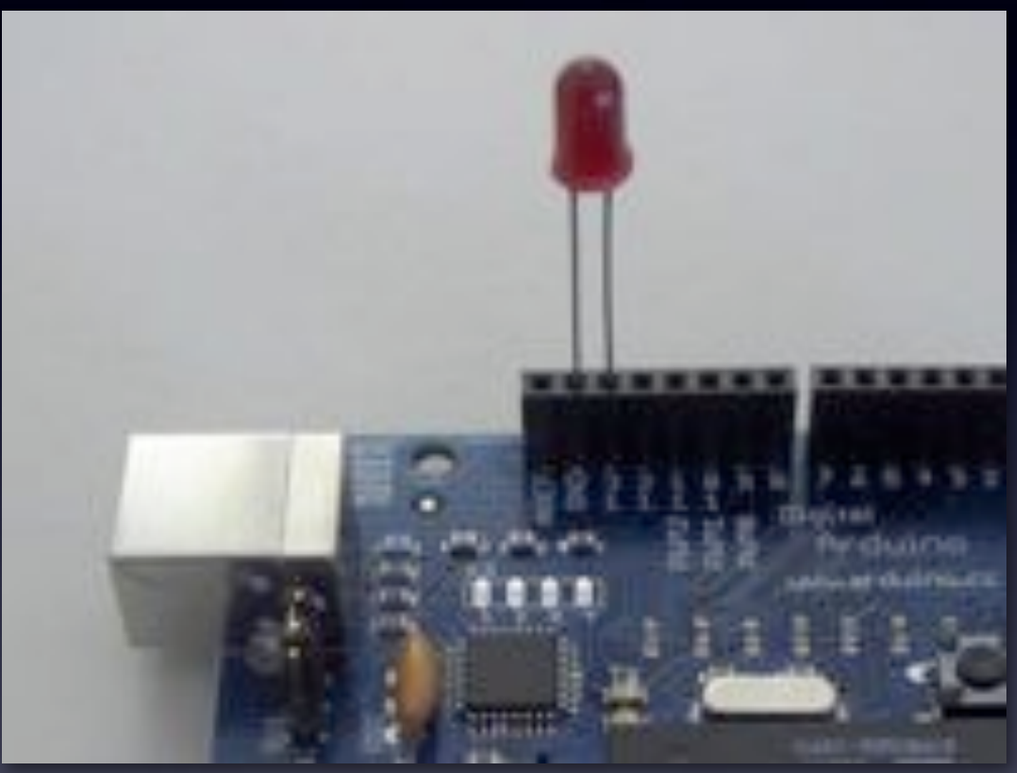
SoCal Code Camp Jan. 27, 2007

Demo



1. Blinky - for "*Hello World*" of Embedded Computing
2. Anlog In - Photo Resistors and Infrared Detectors
3. Interrupt - *LOW, CHANGE, RISING,* and *FALLING*
4. Ultrasonic Sensor - pulseIn
5. fibonacci - Recursive Programming on Arduino ?

"hello world" - blinking LED



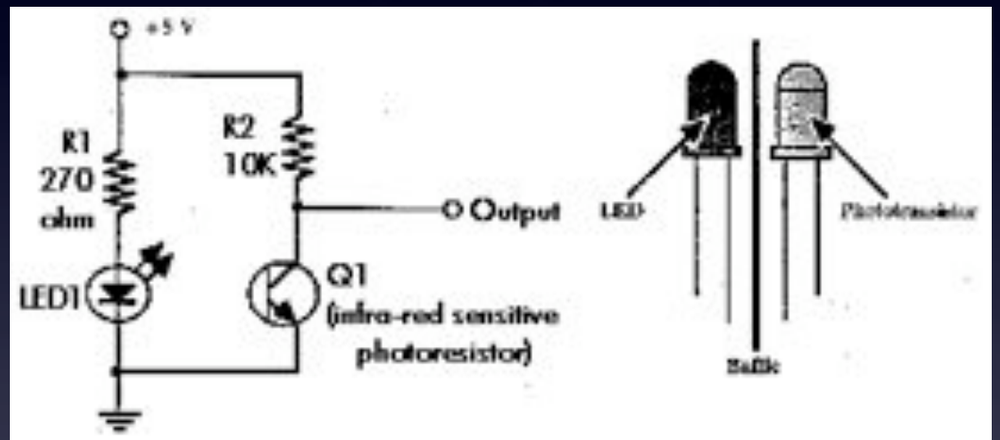
```
int ledPin = 13;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);
  delay(1000);
  digitalWrite(ledPin, LOW);
  delay(1000);
}
```

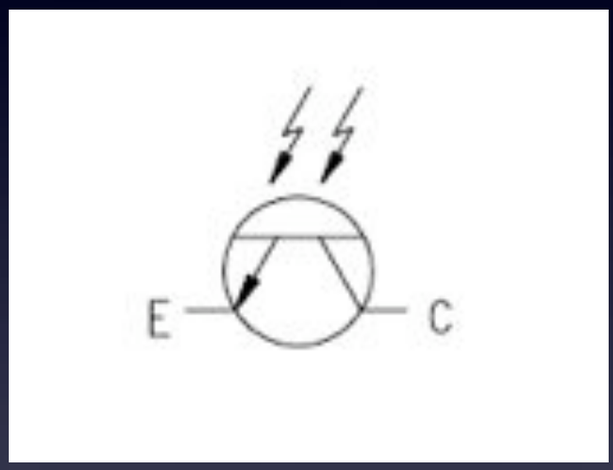
zoom in on next slide

"hello world"

```
int p = 13;           // LED connected to digital pin 13

void setup() {
  pinMode( p,  OUTPUT );   // define digital pin as output
}

void loop() {
  digitalWrite( p,  HIGH );      // sets the LED on
  delay( 1000 );                 // waits for one second
  digitalWrite( p,  LOW );       // sets the LED off
  delay( 1000 );                  // waits for one second
}
```

Blinky - Hello World





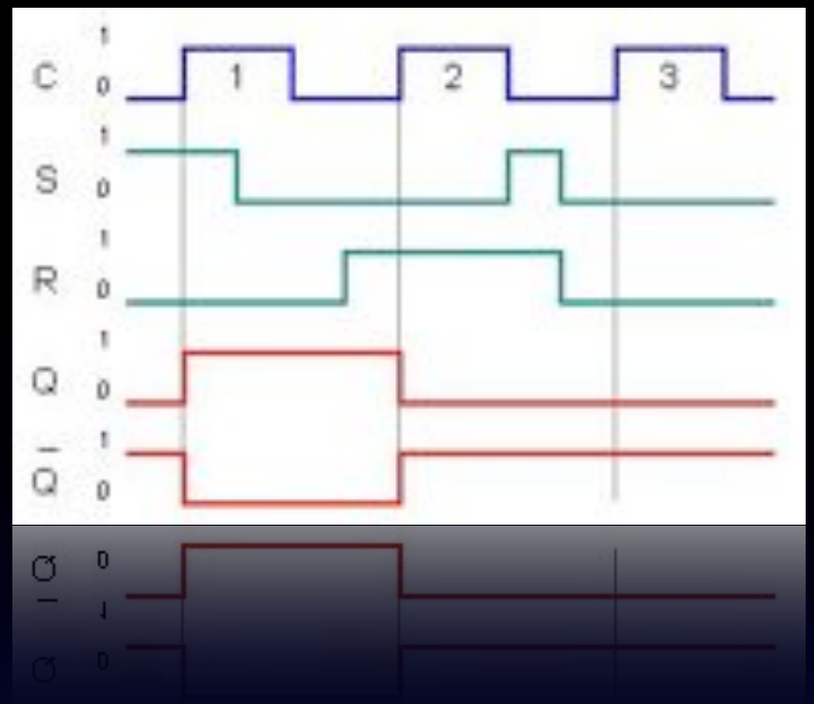
Infrared Detectors

SoCal Code Camp Jan. 27, 2007

```
/*
 *  Code Camp Example: IR on/off switch for external LED
 */
int ledPin = 13;      // LED connected to digital pin 13
int irPin = 1;        // IR receiver connected to analog pin 1
int threshold = 512;  // threshold
boolean b = true;

void setup() {
  pinMode(ledPin, OUTPUT);   // sets the digital pin 13 as output
  Serial.begin(9600);
}
```

```
void irReader() {
  int val = analogRead(irPin);
  if (val<900) {
    b=!b;
  }
  delay(250);
  Serial.println(val, DEC);
}

void loop() {
  irReader();
  digitalWrite(ledPin, b);
}
```

Infrared Detectors



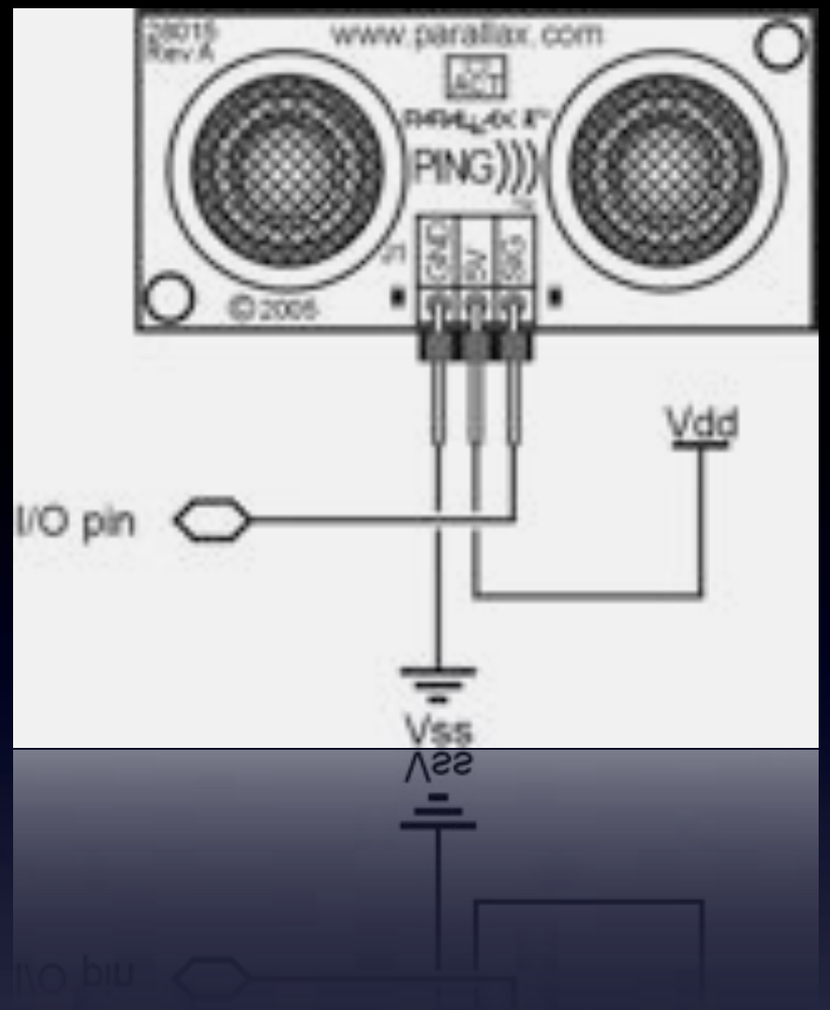
- LOW: interrupt is triggered, whenever the pin is low
- CHANGE: interrupt is triggered whenever the pin changes value,
- RISING: interrupt is triggered, when the pin goes from low to high
- FALLING: interrupt is triggered when the pin goes from high to low

External Interrupts

```
int sensePin = 2;
int pin = 13;
int state = LOW;

void setup(){
  pinMode(sensePin, INPUT);
  pinMode(pin, OUTPUT);
  attachInterrupt(0, blink, CHANGE);
}

void loop(){}

void blink(){
  state = !state;
  digitalWrite(pin, state);
}
```

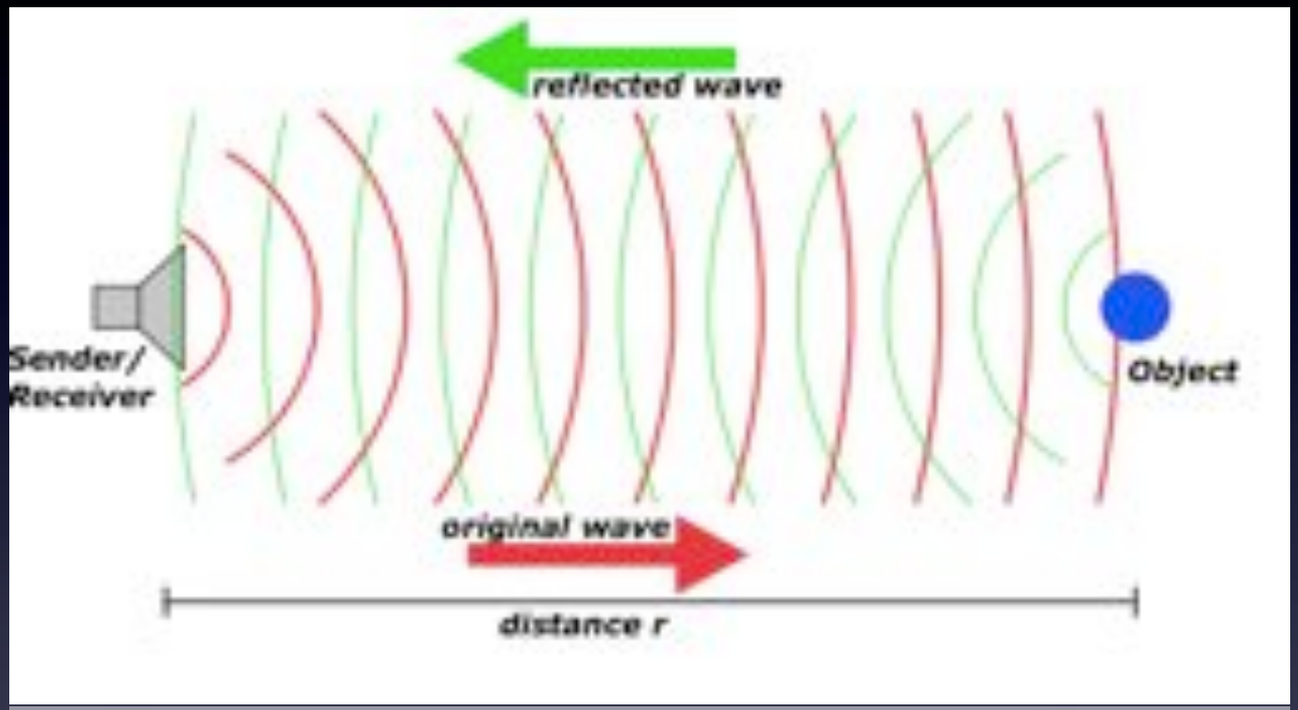
External Interrupts

```
int pin = 7;
unsigned long duration;

void setup() {
  pinMode(pin, INPUT);
}

void loop() {
  duration = pulseIn(pin, HIGH);
}
```



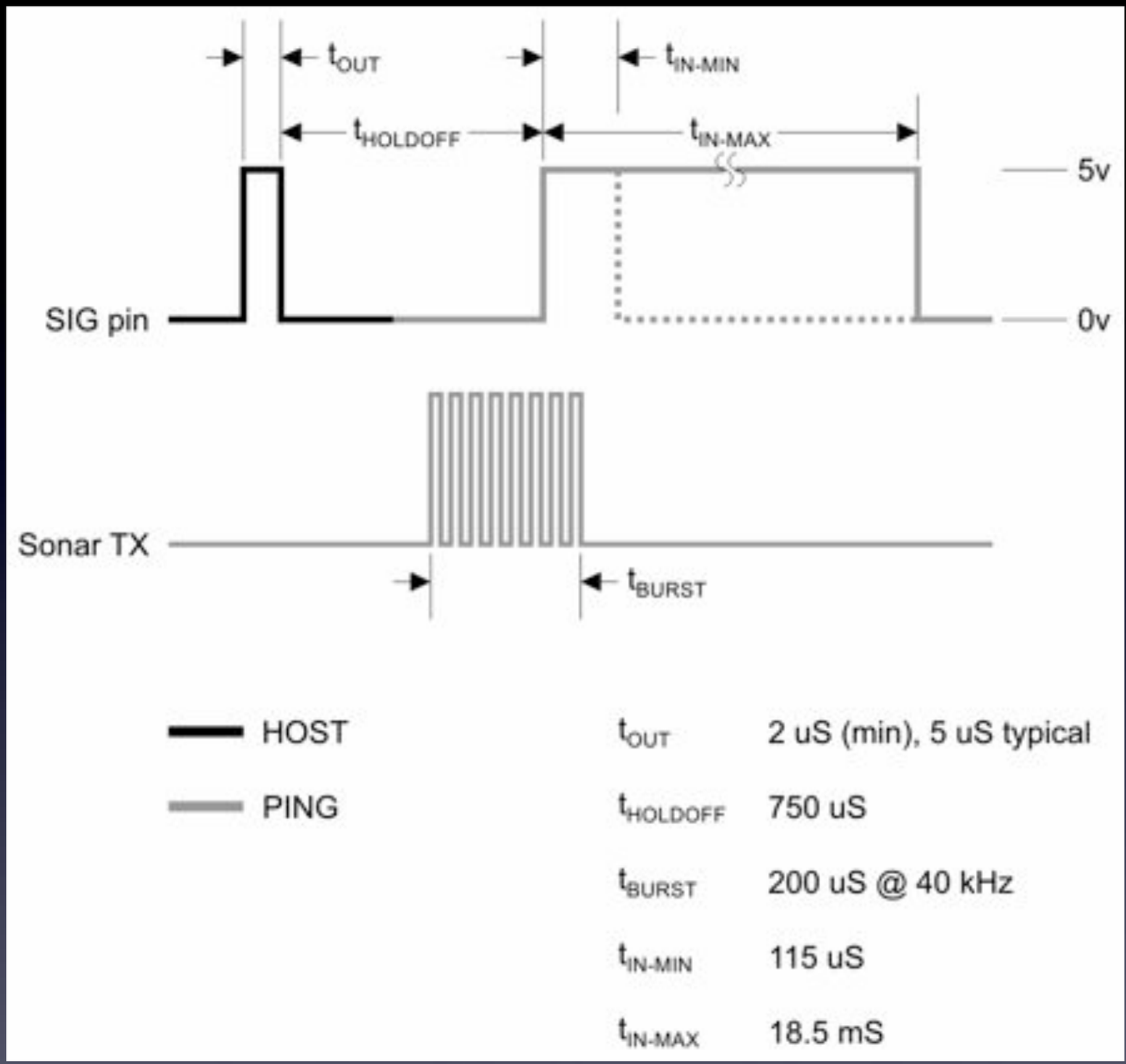
pulseIn

At sea level, at a temperature of 21 °C (70 °F) and under normal atmospheric conditions, the speed of sound is 344 m/s (770 mph).



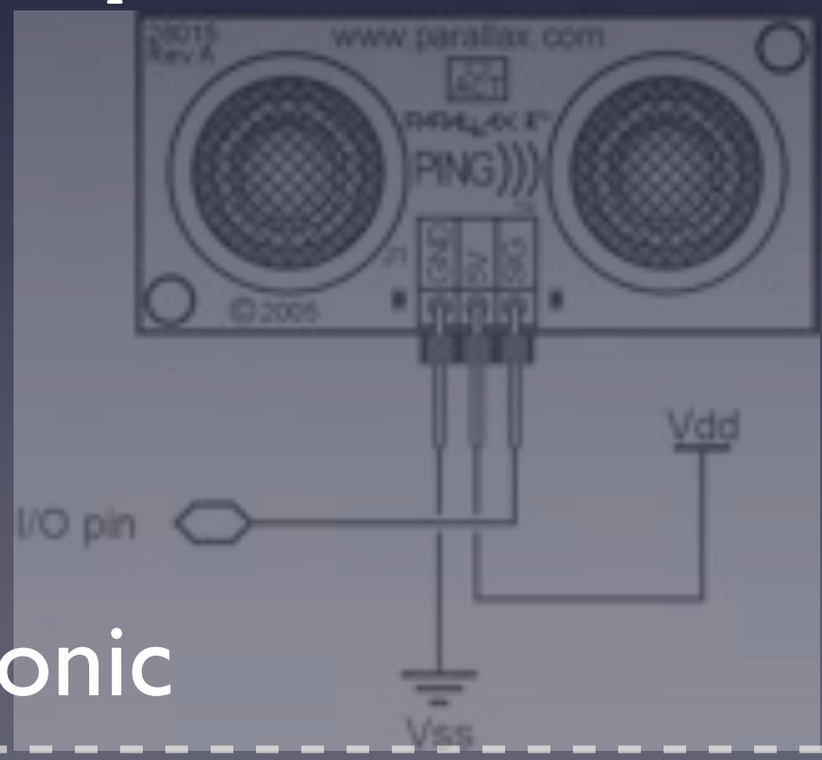
SONAR (SOund Navigation And Ranging)

SoCal Code Camp Jan. 27, 2007



SONAR (SOund Navigation And Ranging)

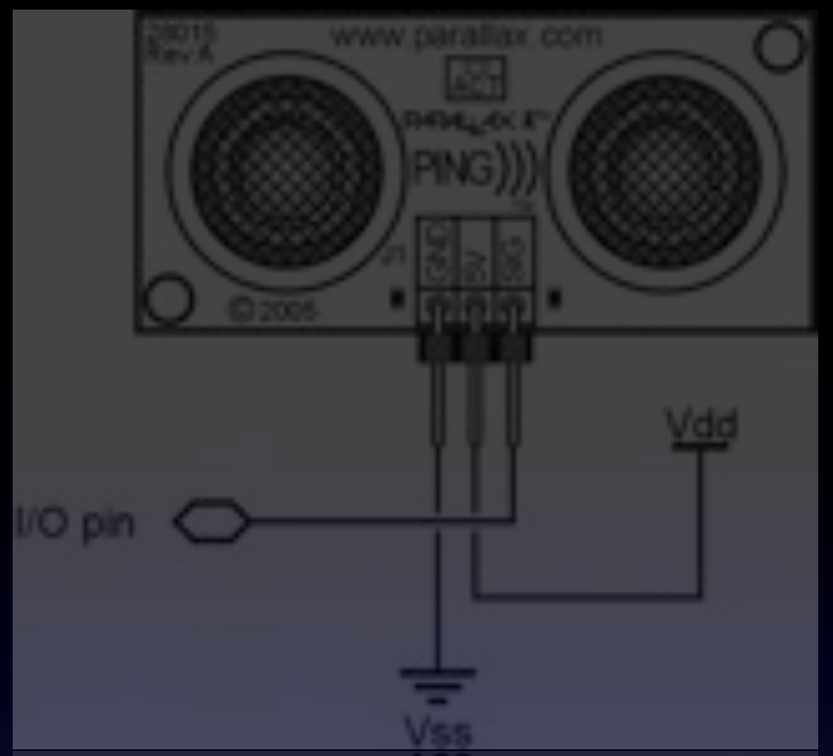
```
int pingPort = 12;                // Ultrasound signal pin

void setup() {
  Serial.begin(9600);             // Sets the baud rate to 9600
}

void squareWave() {
  pinMode(pingPort, OUTPUT);      // Switch signalpin to output

  digitalWrite(pingPort, LOW);    // Send a square wave low high low
  delayMicroseconds(5);           // Wait for 2 microseconds
  digitalWrite(pingPort, HIGH);   // Send high pulse
  delayMicroseconds(5);           // Wait for 5 microseconds
  digitalWrite(pingPort, LOW);
  delayMicroseconds(200);         // Wait for 200 microseconds (holdoff)

  pinMode(pingPort, INPUT);       // Switch signalpin to input
}

void loop() {

  squareWave();
  unsigned long dt= pulseIn(pingPort, HIGH) / 63 ;

  Serial.print("cm=");
  Serial.println(dt);
  Serial.println("");
  delay(500);
}
```



UltraSonic


```
int pingPort = 12;              // Ultrasound signal pin

void setup() {
  Serial.begin(9600);           // Sets the baud rate to 9600
}
```

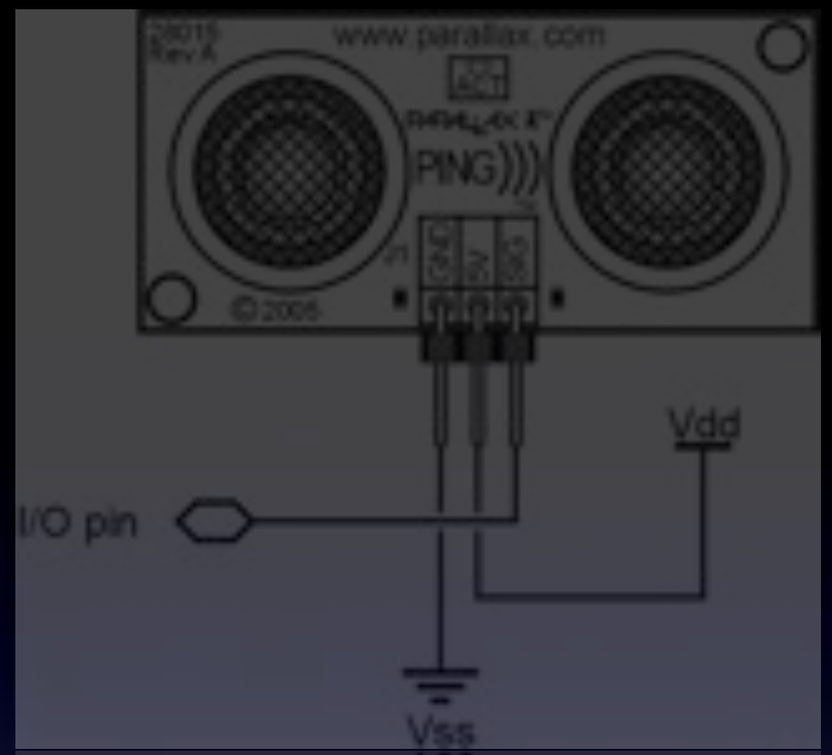
UltraSonic




```
void squareWave() {
  pinMode(pingPort, OUTPUT);     // Switch signalpin to output

  digitalWrite(pingPort, LOW);   // Send a square wave low high low
  delayMicroseconds(5);          // Wait for 2 microseconds
  digitalWrite(pingPort, HIGH); // Send high pulse
  delayMicroseconds(5);          // Wait for 5 microseconds
  digitalWrite(pingPort, LOW);
  delayMicroseconds(200);        // Wait for 200 microseconds (holdoff)

  pinMode(pingPort, INPUT);      // Switch signalpin to input
}
```

UltraSonic

At sea level, at a temperature of 21 °C (70 °F) and under normal atmospheric conditions, the speed of sound is 344 m/s (770 mph). This equates to 1 cm in 29.069 uS.




```
  squareWave();
  unsigned long dt= pulseIn(pingPort, HIGH) / 63 ;
  // theoretically we should divide by 58 (round trip)
  Serial.print("cm=");
  Serial.println(dt);
  Serial.println("");
  delay(500);
}
```

UltraSonic

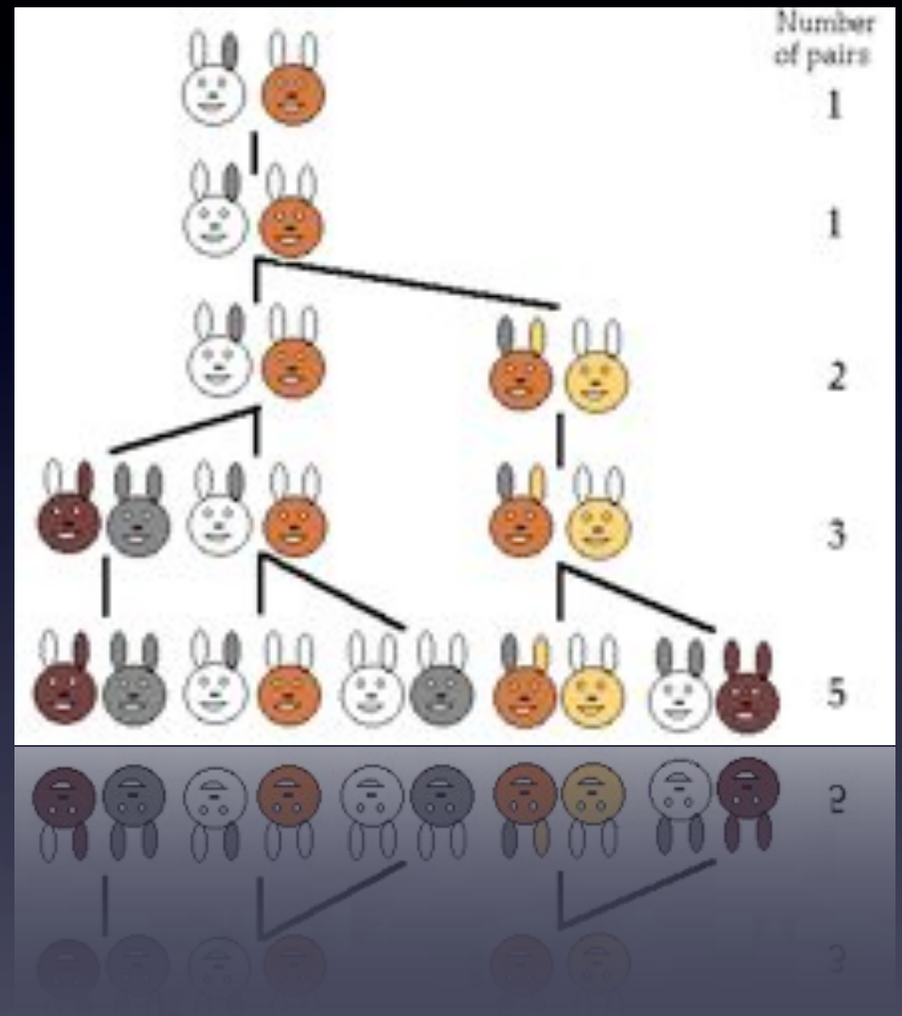
Fibonacci Number

$$F(n) := \begin{cases} 0 & \text{if } n = 0; \\ 1 & \text{if } n = 1; \\ F(n-1) + F(n-2) & \text{if } n > 1. \end{cases}$$

15,000,000,000

0

1	1
2	1
3	2
4	3
5	5
6	8
7	13
8	21
9	34
10	55
11	89
12	144
13	233
14	377
15	610
16	987
17	1597
18	2584
19	4181
20	6765
21	10946
22	17711
23	28657
24	46368
25	75025
26	121393
27	196418
28	317811
29	514229
30	832040
31	1346269
32	2178309
33	3524578
34	5702887
35	9227465
36	14930352
37	24157817
38	39088169
39	63245986
40	102334155
41	165580141
42	267914296
43	433494437
44	701408733
45	1134903170
46	1836311903
47	2971215073
48	4807526976
49	7778742049
50	12586269025



Fibonacci Numbers

SoCal Code Camp Jan. 27, 2007

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

unsigned long fib(int n) {
  if (n<=1)
    return n;
  else
    return fib(n-1) + fib(n-2);
}
```

zoom in on next slide

```
void loop(void){
  for(int i=0; i<100; i++) {
    Serial.print( "fib(" );
    Serial.print( i, DEC );
    Serial.print( ")= " );
    Serial.println( fib(i), DEC);
  }
  delay(20000);
} //repeat forever
```

Fibonacci Numbers

Fibonacci Number

```
unsigned long fib(int n) {
   if (n<=1)
      return n;
   else
      return fib(n-1) + fib(n-2);
}
```

Fibonacci Numbers

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

unsigned long sum(int n) {
  if (n<=1)
    return n;
  else
    return n + sum(n-1);
}

void loop(void){
  for(int i=0; i<1000; i++) {
    unsigned long s = sum(i);
    unsigned long t = (i+1L)*i/2L;
    if (s == t) {
      Serial.print("sum(");
      Serial.print(i,DEC);
      Serial.print(")= ");
      Serial.println(s,DEC);
    } else {
      Serial.println("error");
      break;
    }
  }
  delay(20000);
} //repeat forever
```

zoom in on next slide

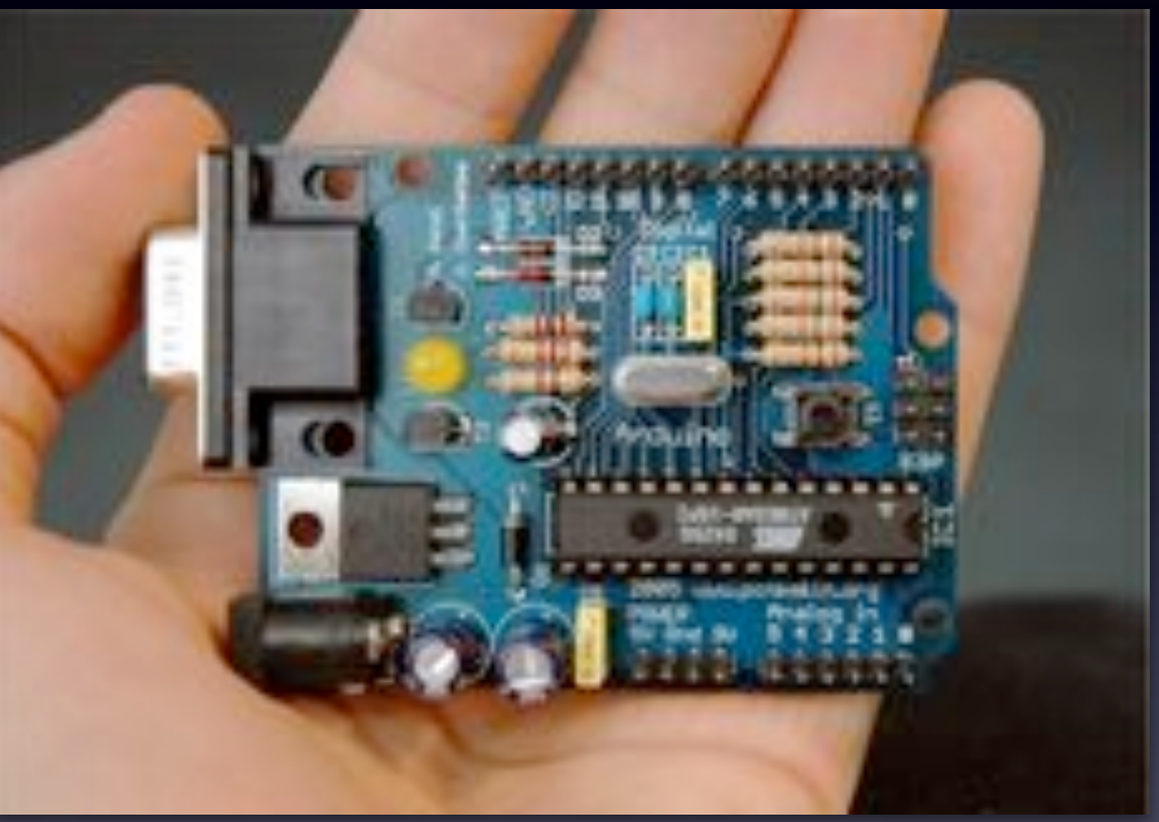
Sum 1..n = $n(n+1)/2$

Sum

$$\sum_{i=1}^{n} i = 1 + 2 + ... + n$$

```
unsigned long sum(int n) {
   if (n<=1)
      return n;
   else
      return n + sum(n-1);
}
```

Sum 1..n = $n(n+1)/2$



Has pervasive, ubiquitous computing finally arrived?

ubiquitous
present, appearing, found everywhere
pervasive
spreading widely throughout an area or a group of people

Thanks for coming