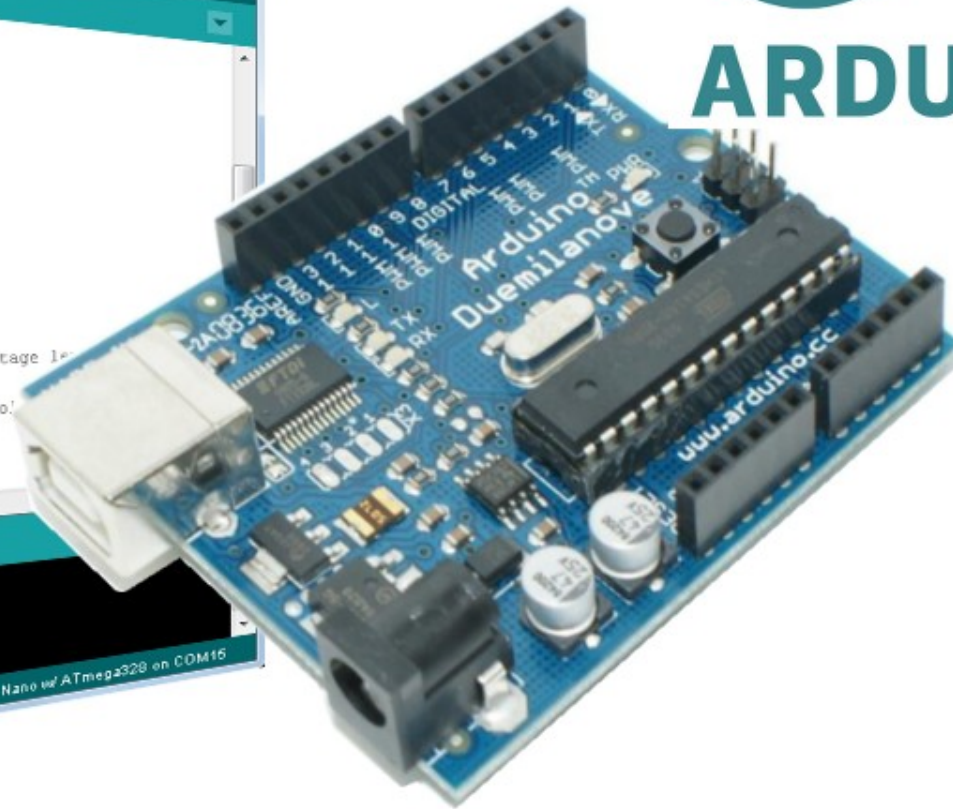
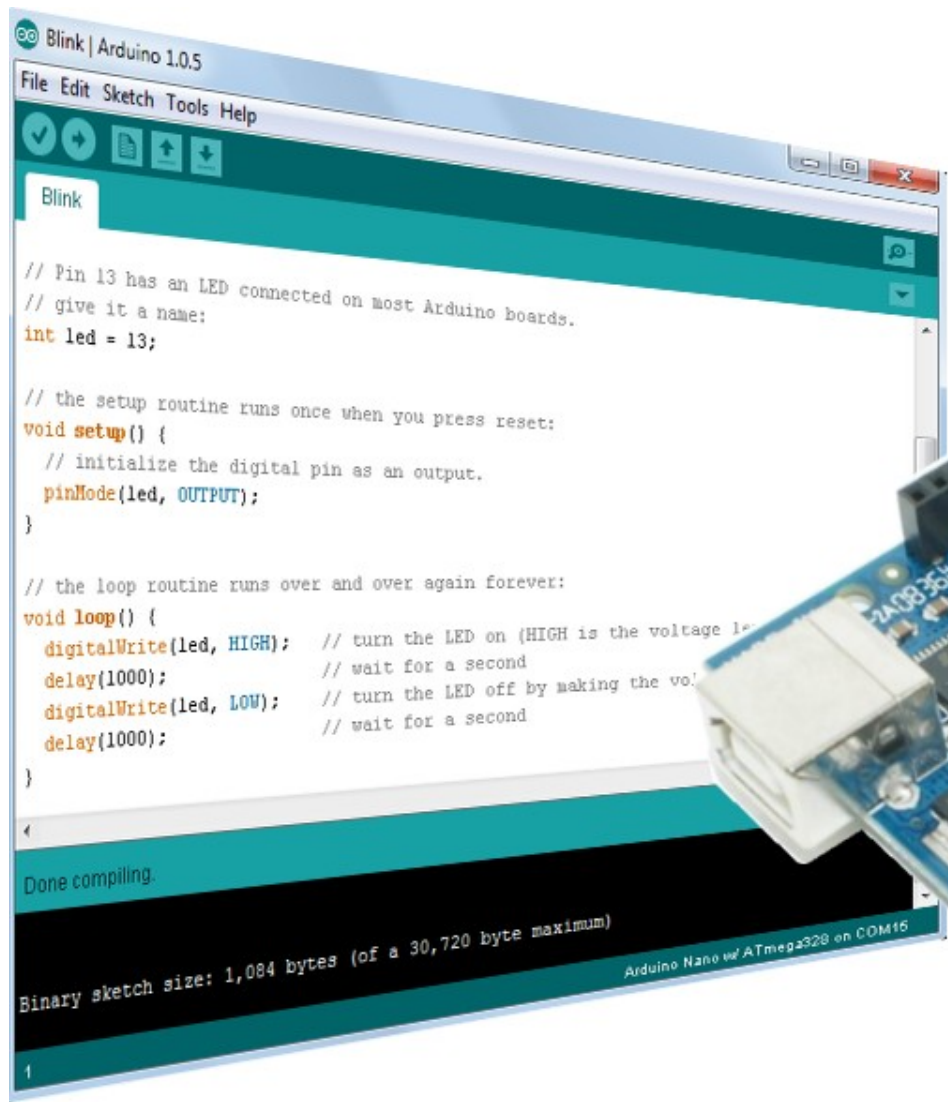


# Arduino tanfolyam kezdőknek és haladóknak



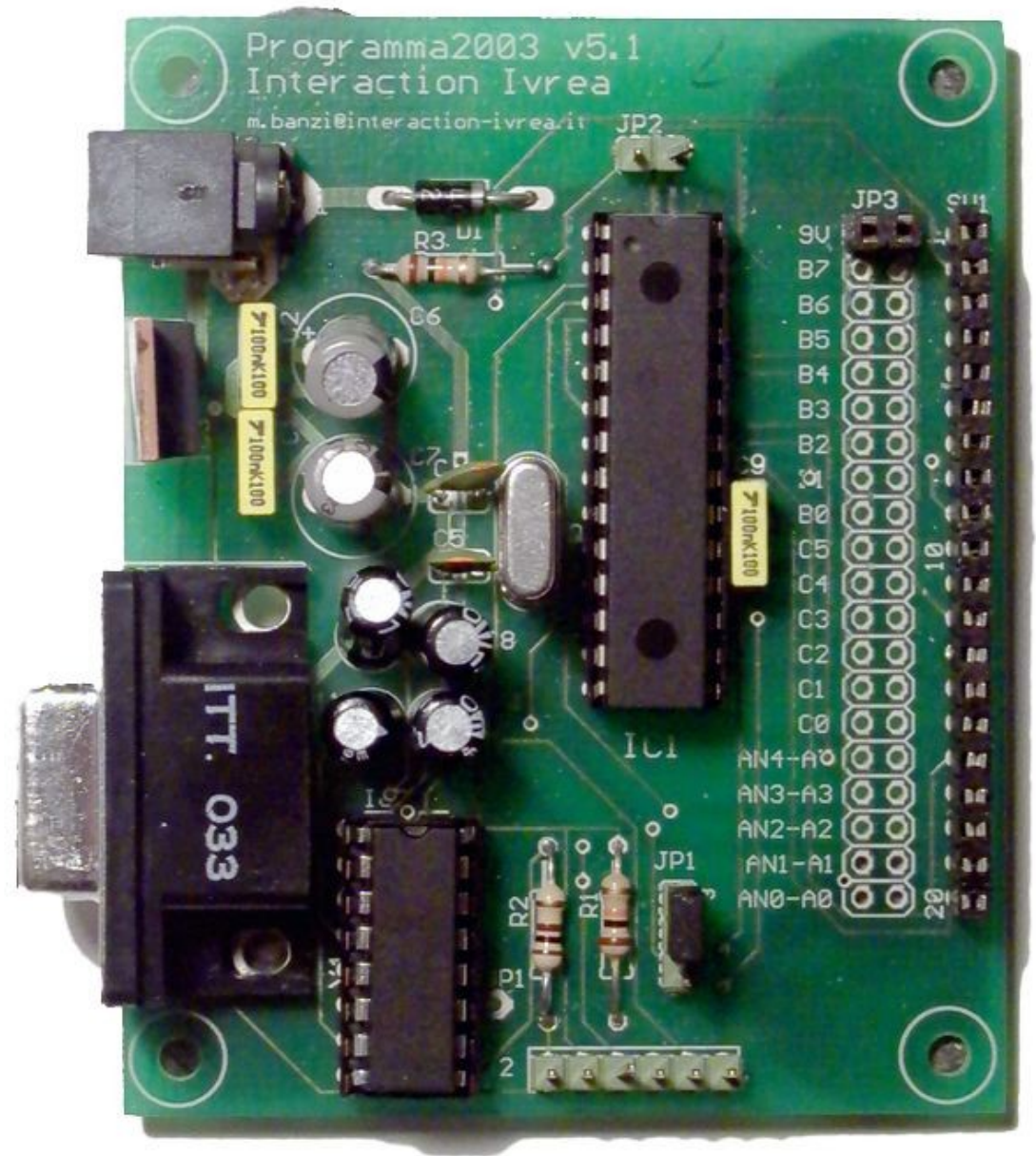
## 7. Processing, Wiring, Arduino – ahogy elkezdődött

# Ivrea, az Arduino szülővárosa

- Az **Interaction Design Institute, Ivrea (IDII)** diplomás képzést végzett az Interaction Design területén 2001-2005 között, Ivreában, Olaszország északi részén
- **Interaction design** – a felhasználó és a termék közötti interakció megtervezése (kezelőszervek, kijelzők, menük, hangjelzés stb.)
- *”In the same way that industrial designers have shaped our everyday life through objects that they design for our offices and for our homes, interaction design is shaping our life with interactive technologies – computers, telecommunications, mobile phones, and so on. If I were to sum up interaction design in a sentence, I would say that it’s about shaping our everyday life through digital artifacts – for work, for play, and for entertainment.”*  
— Gillian Crampton Smith, interview of January 30, 2002
- Az interakció tervezés modellezéséhez, prototípus készítéshez könnyen kezelhető és olcsó **mikrovezérlős fejlesztői rendszert** kerestek, ennek során született meg a **Wiring**, majd az **Arduino**

# Programma2003: az első próbálkozás

- Massimo Banzi, az IDII tanár  
PIC16F876 MCU (28 pin)  
Soros porti csatlakozás  
Analóg bemenetek  
Digitális I/O
- JAL fordító ver 1.0 2003  
(Wouter van Ooijen)
- Problémák:
  - ❖ Microchip (a gyártó) nem ad szabadszoftvert, egyébként is csak assemblert biztosított
  - ❖ A JAL fordító akkor még elég kezdetleges volt, nulla támogatással



# A Wiring koncepció

- **Hernando Barragán** (témavezetői: Massimo Banzi és Casey Reas) diplomamunkája volt a **Wiring** koncepció kidolgozása és megvalósítása: [Wiring: Prototyping Physical Interaction Design](#)
- Egyszerű integrált fejlesztői környezet, a **Processing**-re alapozva
- Egyszerű nyelv vagy keretrendszer a mikrovezérlőkhöz
- Integrált fejlesztői eszközök (a felhasználónak transzparens módon)
- Bootloader a programok könnyű letöltéséhez
- Soros monitor a mikrovezérlő ellenőrzésére és kétoldalú adatátvitelhez
- Nyíltforrású szoftver
- Nyíltforrású hardver terv Atmel mikrovezérlőre alapozva
- Széleskörű online referencia a parancsokról, a könyvtárakról, mintapéldák, tananyagok, fórum és Wiring projektek bemutatása

# A Wiring nyelv

- Az **Arduino** környezetben megszokott, általunk is használt utasítások eredetileg a **Wiring** nyelv részeként születtek meg
  - ❖ `pinMode()`
  - ❖ `digitalRead()`
  - ❖ `digitalWrite()`
  - ❖ `analogRead()`
  - ❖ `analogWrite()`
  - ❖ `delay()`
  - ❖ `millis()`
- Fontos döntés volt a kivezetések absztrakciója (sorszámozás)
- A nyelv a hardver fejlesztéssel párhuzamosan még alakult, például a `digitalWrite()` az eredeti elképzelés szerint még `pinWrite()` volt

Hernano Barragán:

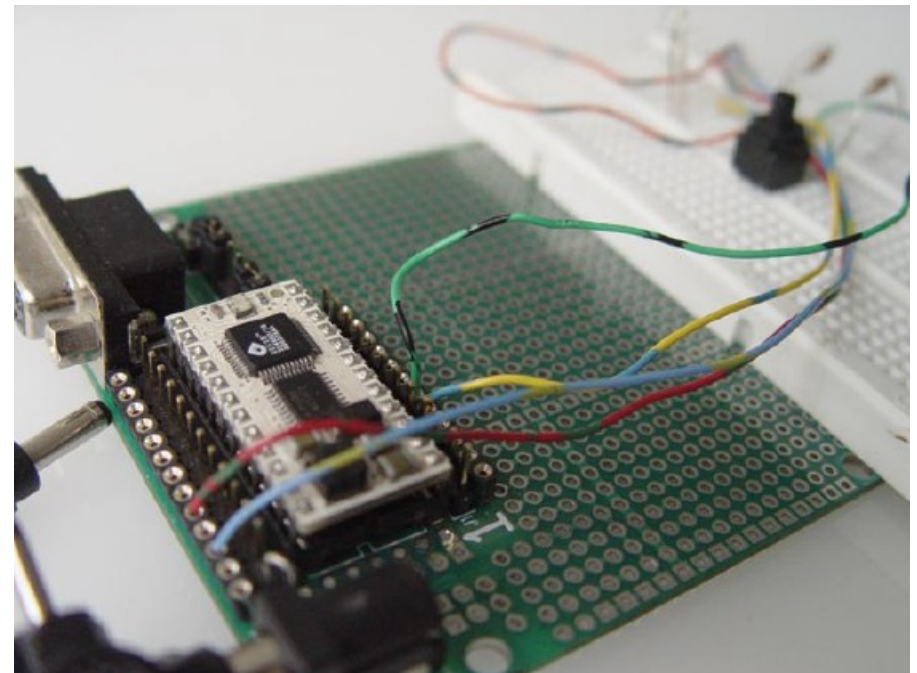
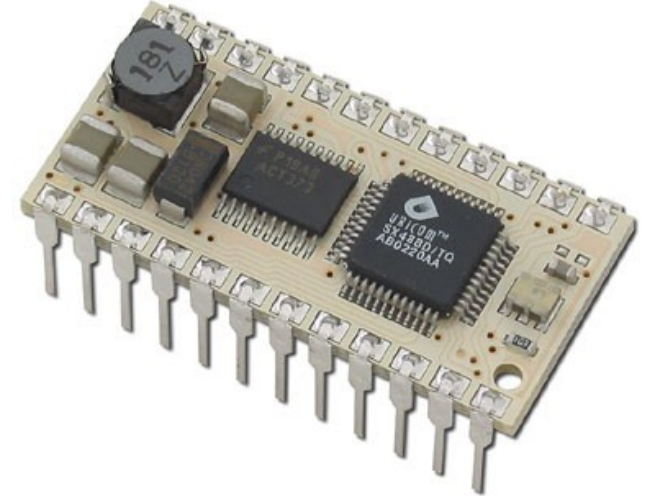
[Wiring: Prototyping Physical Interaction Design](#)

Hernando Barragán:

[The Untold History of Arduino](#)

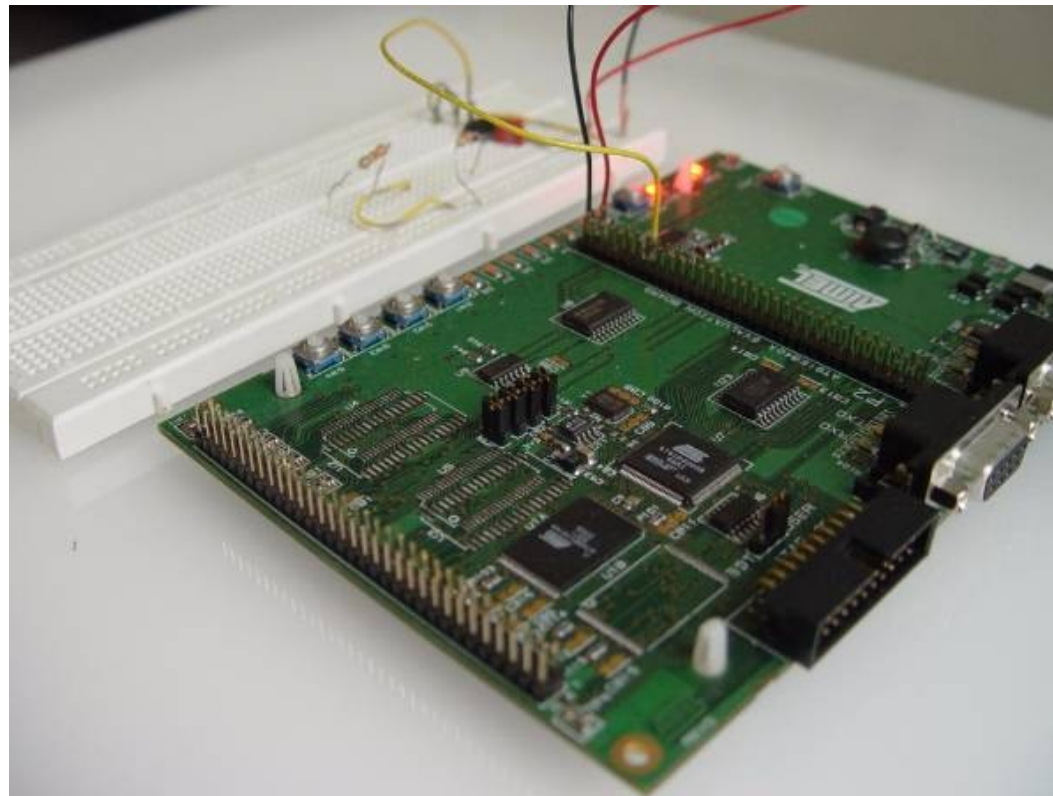
# A hardver tervezése

- A fejlesztés legnehezebb része a hardver kiválasztása és kialakítása
- **1. Prototípus:** Parallax Javelin Stamp
- Mivel a **Processing** IDE Java alapú kódot fordít, kézenfekvő volt egy (lebutított) Java nyelven programozható eszköz választása
- **Probléma:** A Javelin Stamp kártyára írt programok csak a gyártó szoftverével fordíthatók, tölthetők le, így a nyíltforrású követelménynek nem felel meg



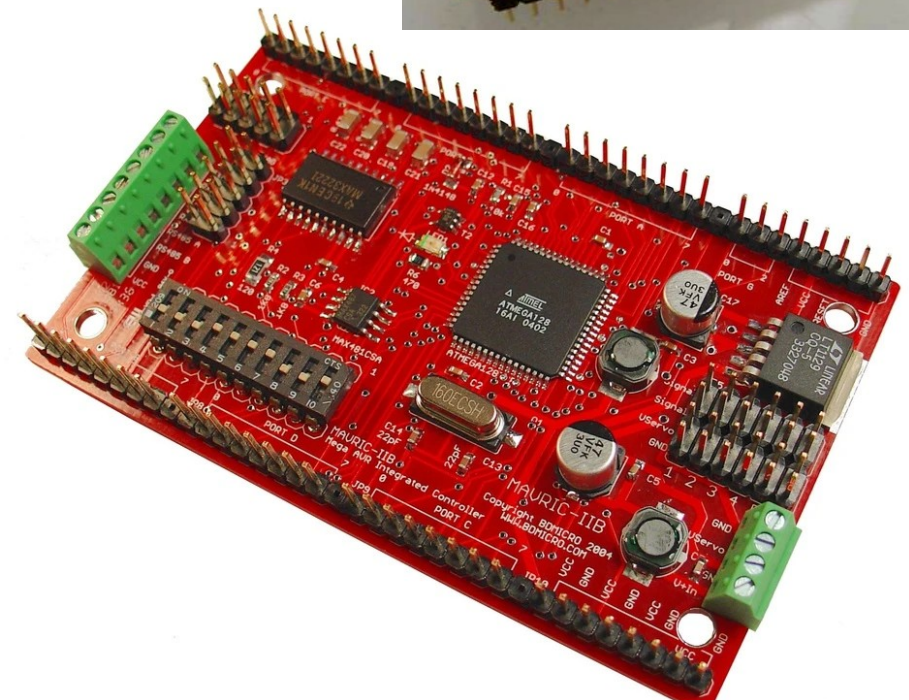
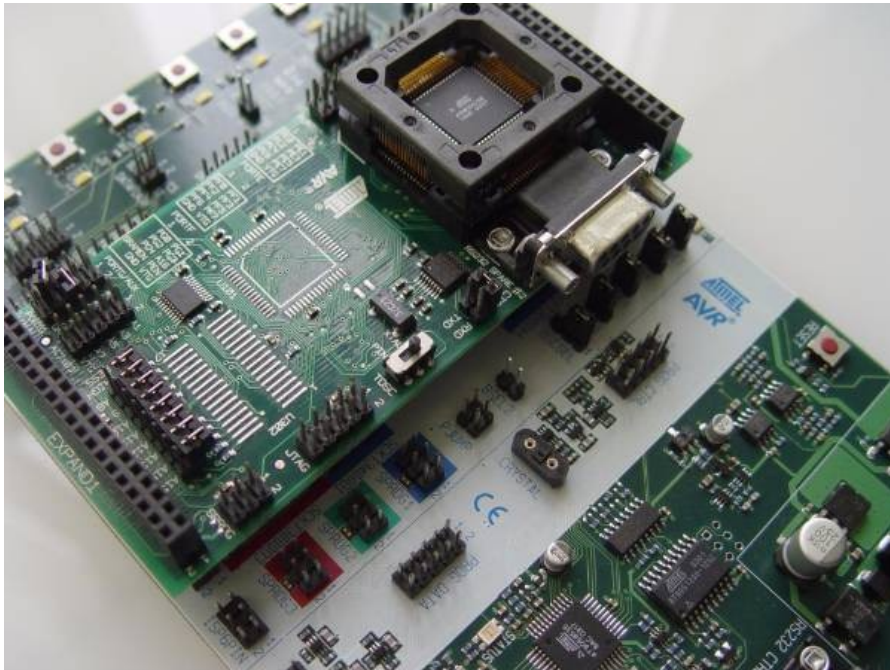
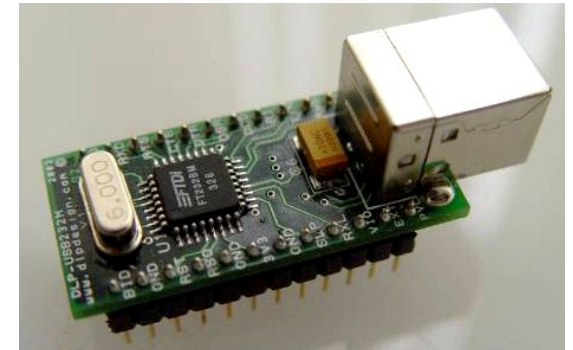
## 2. Prototípus: Atmel AT91R40 008

- Az ARM7TDMI alapú AT91R40 008 32-bites nagy teljesítményű mikrovezérlő külső (flash) memóriát kezel és 256 kilobájt belső SRAM memóriát tartalmaz. Az órajel 0 – 75 MHz lehet.
- **Probléma:** túl bonyolult hardver, kézzel nem forrasztható (100 lábú) tokozás, az egyszerű kezelhetőség követelményének nem felel meg



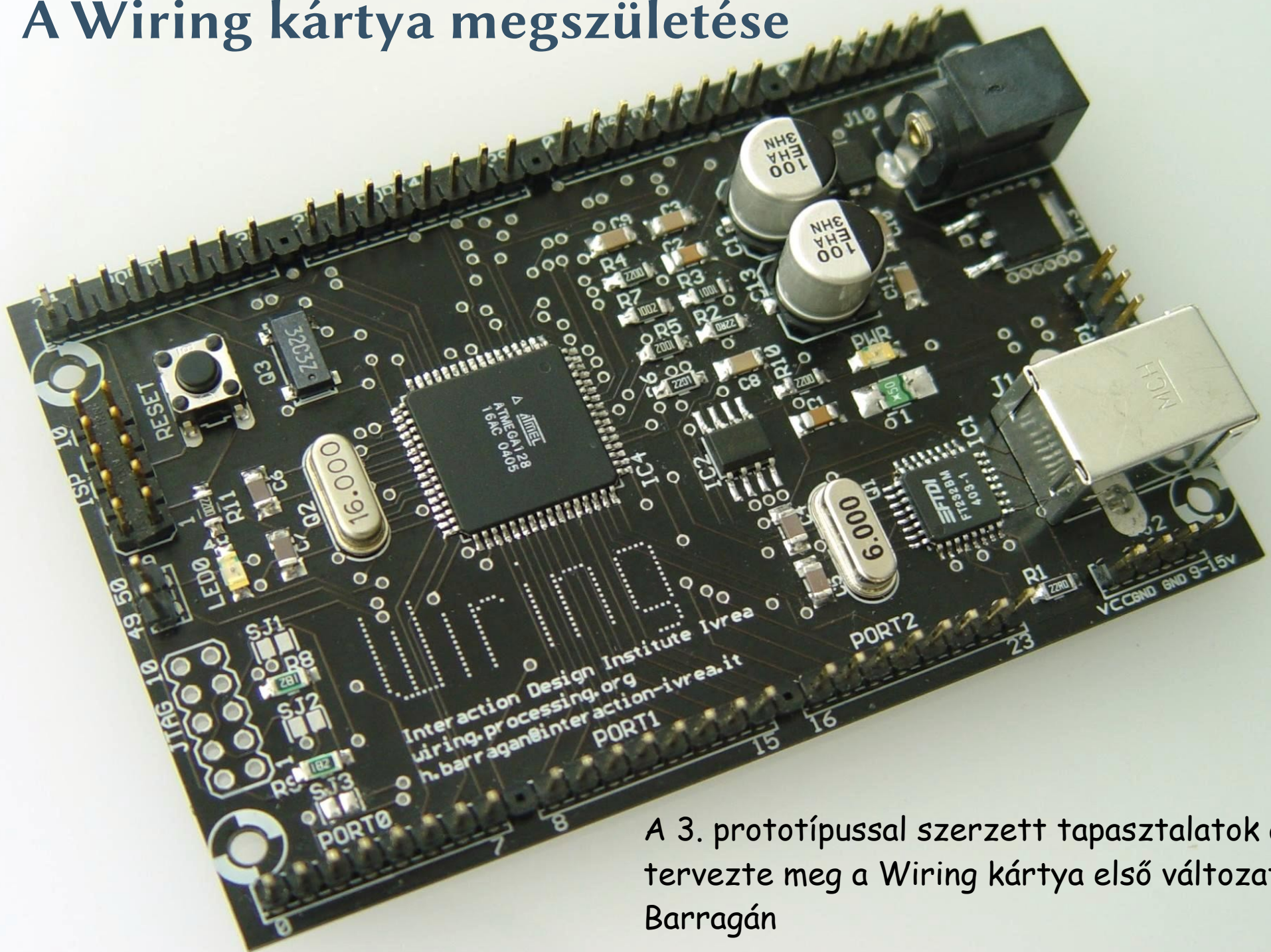
# 3. prototípus

- Az előzőnél kisebb teljesítményű, egyszerűbb processzorok közül az **ATmega128**-ra esett a választás, amely az **STK500** fejlesztői kártya, illetve a **BDMICRO MAVRIC-II** kártya segítségével lett kipróbálva, **FT232** USB-TTL konverterrel
- **Konklúzió:** jól használható





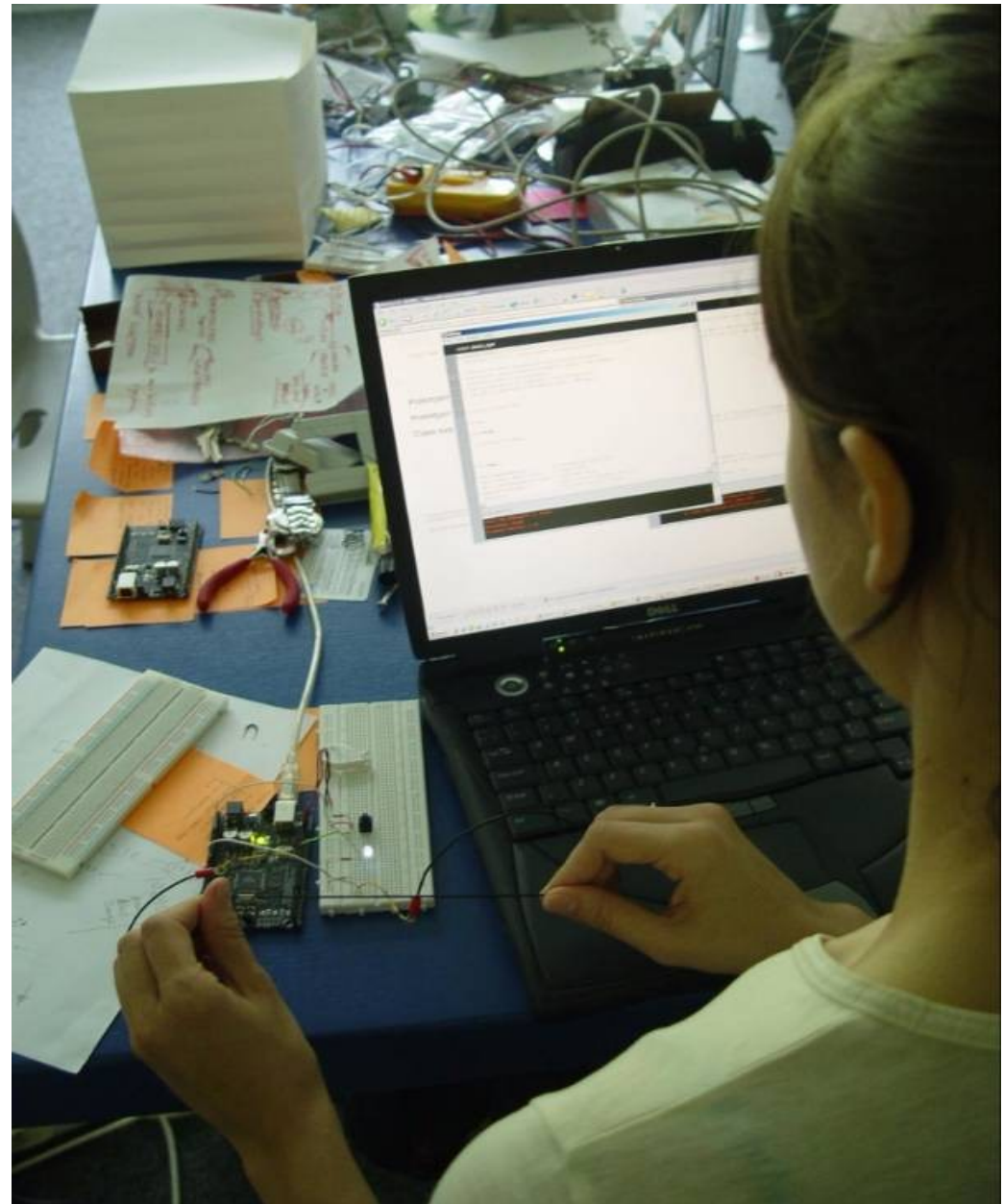
# A Wiring kártya megszületése



A 3. prototípussal szerzett tapasztalatok alapján tervezte meg a Wiring kártya első változatát H. Barragán

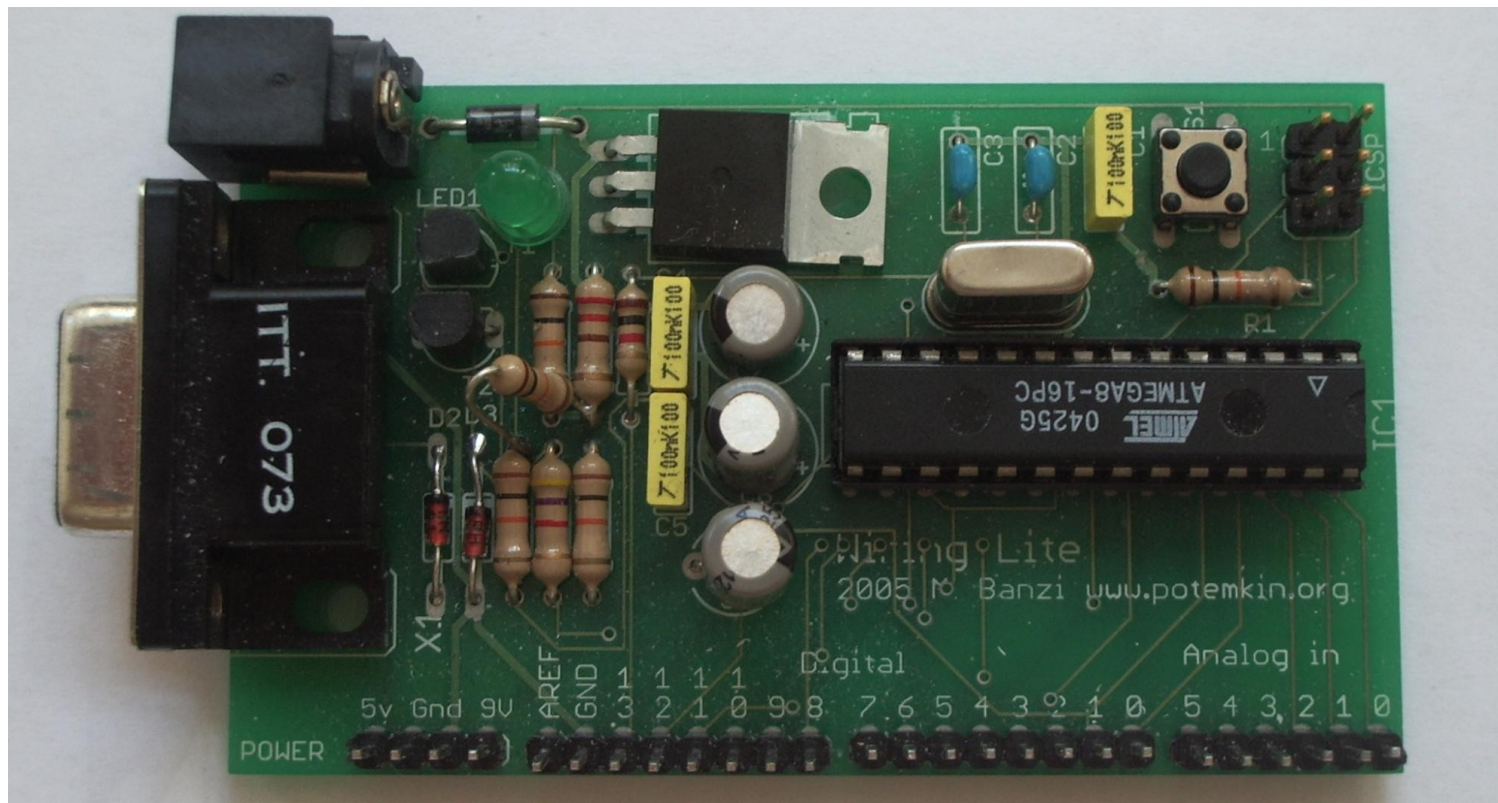
# A Wiring projekt sikere

- A **Wiring** kártya és a fejlesztői környezet sikeres lett, 2004 őszétől már használták a képzésben
- A **Wiring** továbbfejlesztése azonban elszakadt Ivreától, mivel **H. Barragán** hazatért Kolumbiába (Universidad de Los Andes)
- Wiring honlap: [wiring.org.co](http://wiring.org.co)



# Az Arduino megszületése

- Mivel a **Wiring** kártya bonyolultabb és drágább volt a szükségesnél, **Massimo Banzi** tovább dolgozott a hardver egyszerűsítésén
- Az **ATmega8**-alapú kártya még **Wiring-Lite** néven jelent meg, de valójában ez már az első Arduinonak is tekinthető



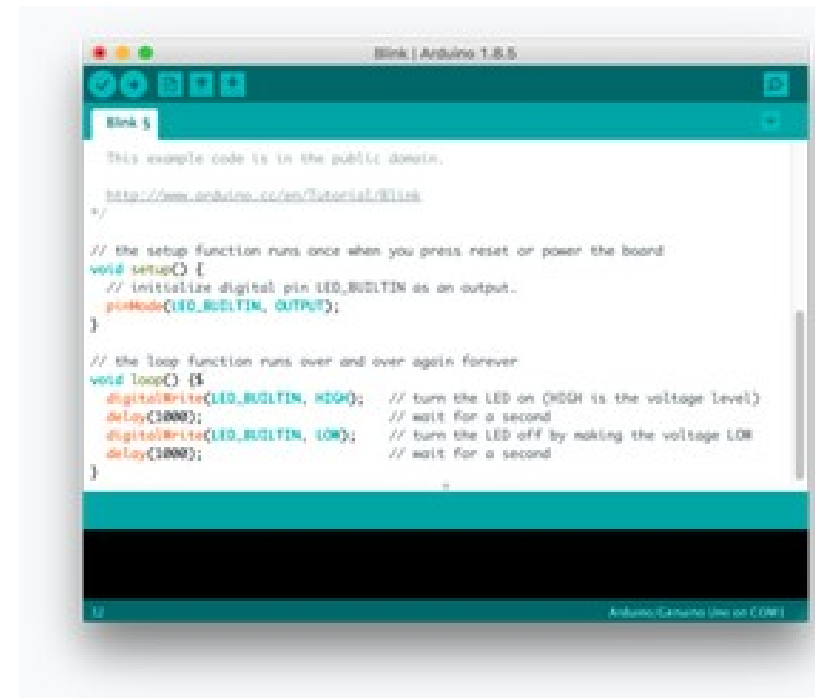
# Az Arduino és a Wiring különválása

- Az **Arduino** fejlesztői környezete a nyíltforrású Wiring „forkja” let, ezzel a két projekt külön vált
- Az **Arduino** fejlesztői csapat:



# Arduino IDE

- **Arduino 00xx** (0023 az utolsó) – ebben még .pde a kiterjesztés
- **Arduino 1.0x** (1.06 az utolsó) – itt már .ino a kiterjesztés, változott az i2c kezelés (wire.h, read és write, a send és receive helyett)
- **Arduino 1.5.xBETA** és **1.6.x** – már nem támogatott
- **Arduino 1.8.x** (jelenleg 1.8.10 a legfrissebb) – moduláris felépítés (ARM, ESP8266 stb. processzorok támogatásával bővíthető, könyvtárak kényelmes telepítése és frissítése)



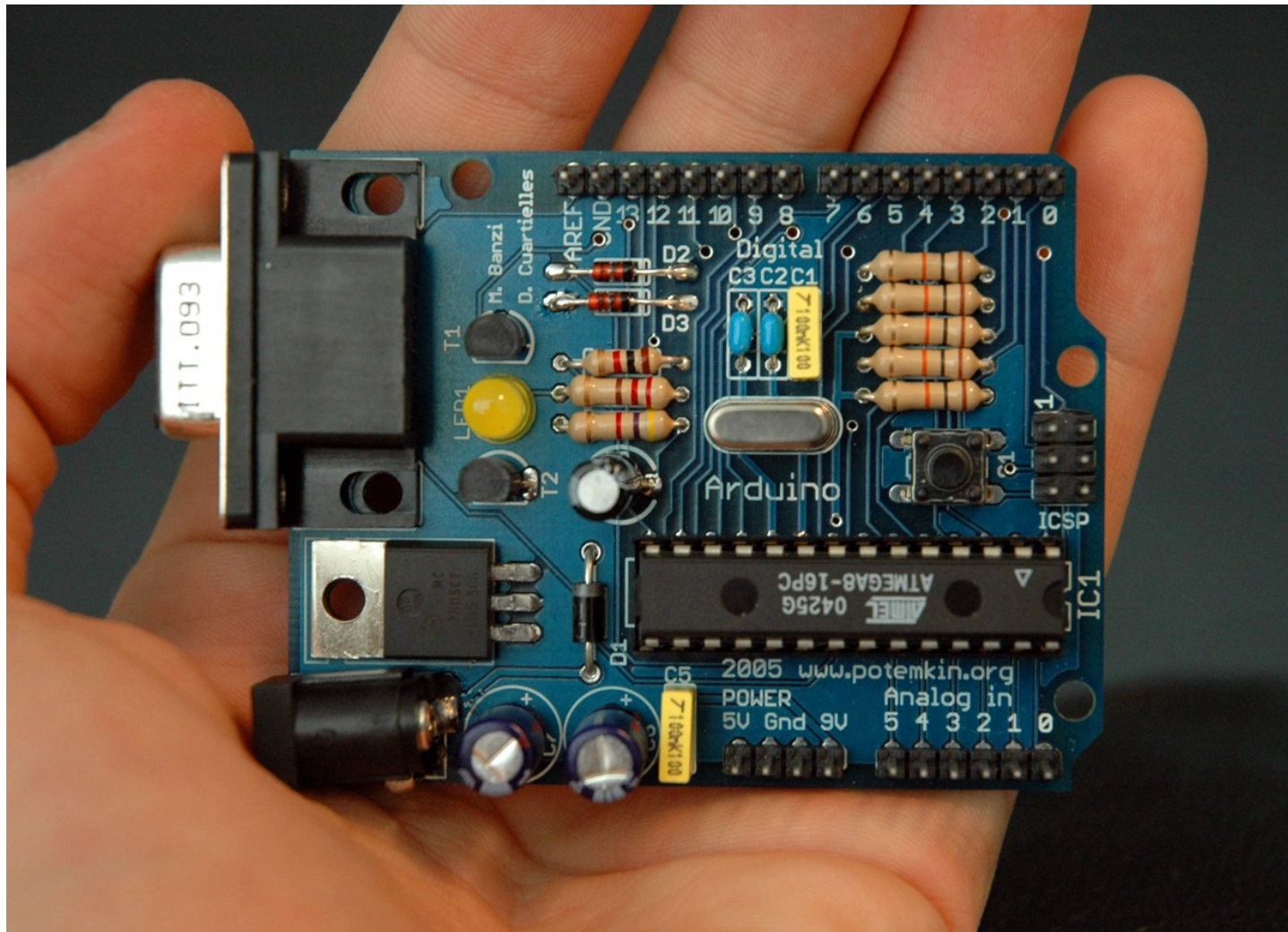
# Honnan van az Arduino elnevezés?

- **Arduino**, olasz nemes, Ivrea örgrófja, 1002-1014 között Itália királya
- Ivreában utcanév és egy kávéház is őrzi az emlékét, amelyben az IDII tanárai szoktak összejönni, innen a névadás ötlete



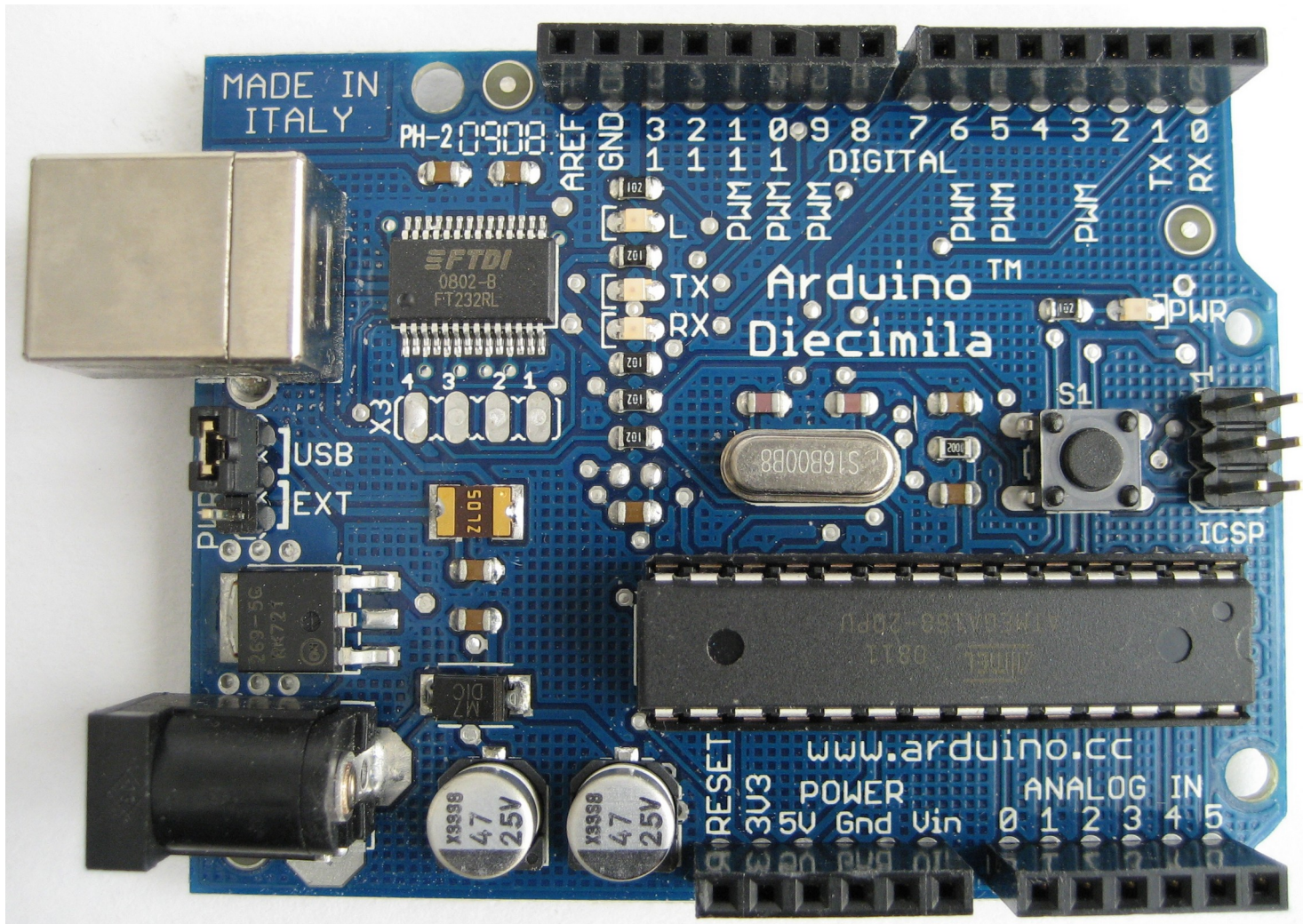
# Arduino RS232 (2005)

- Az első Arduino kártyák még a soros portra csatlakoztak
- ATmega8 vagy ATmega16 mikroprocesszor, 28 lábú tokozás



# Arduino Diecimila

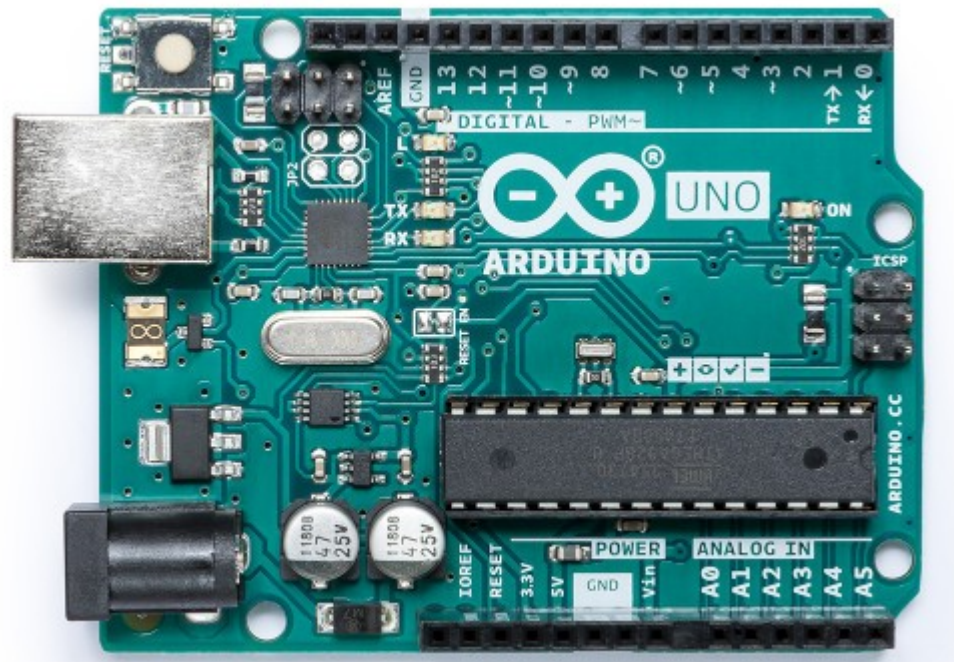
- Az elnevezés azt jelzi, hogy már 10 000 darabot adtak el
- ATmega168 mikrovezérlő, FTDI FT232 USB-UART átalakító





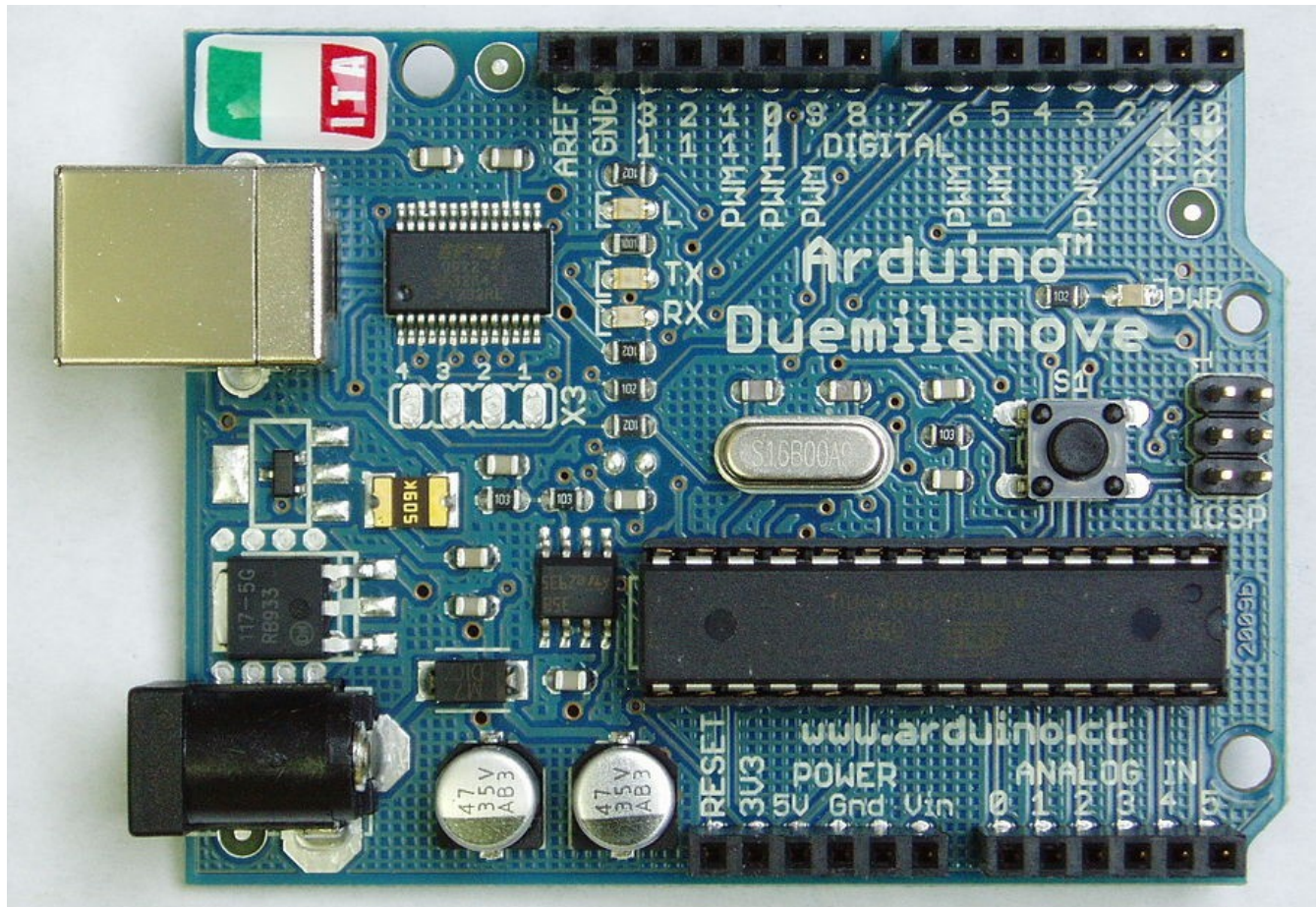
# Arduino UNO

- Az Arduino UNO már mellőzi az FTDI USB-UART átalakítót, helyett egy **ATmega16U2** mikrovezérlő az átalakító
- A kártya **Atmega328P** mikrovezérlővel rendelkezik és új, nagyobb sebességű letöltőprogramot használ
- Megnőtt a kivezetések száma: az I2C busz a felső csatlakozósoron is ki van vezetve



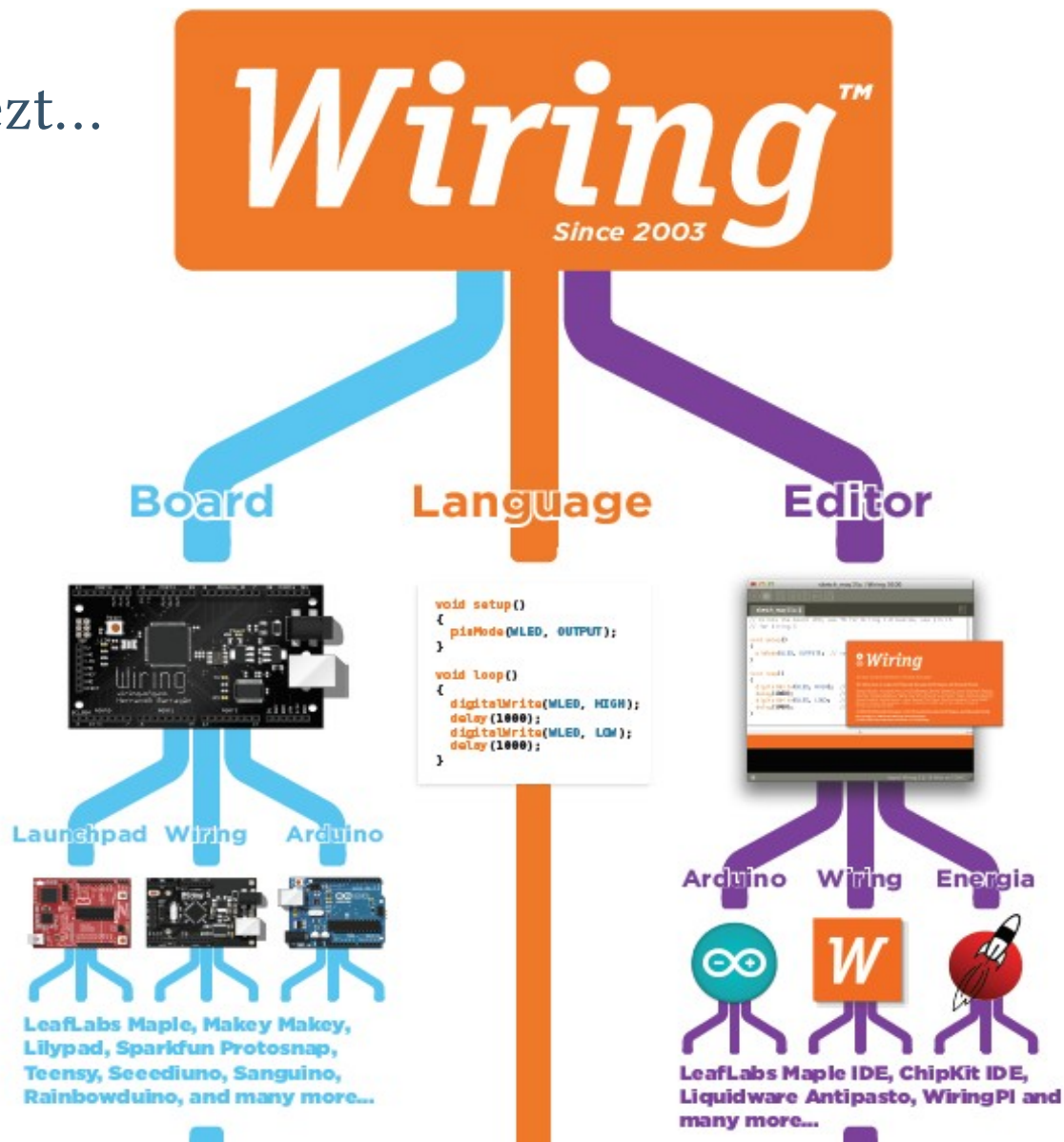
# Arduino Duemilanove

- A név itt a kiadás 2009-es dátumára utal
- Az MCU ATmega168 vagy ATmega328, újdonsága az automatikus tápfeszültség átkapcsoló (FET) a korábbi jumper kiváltására



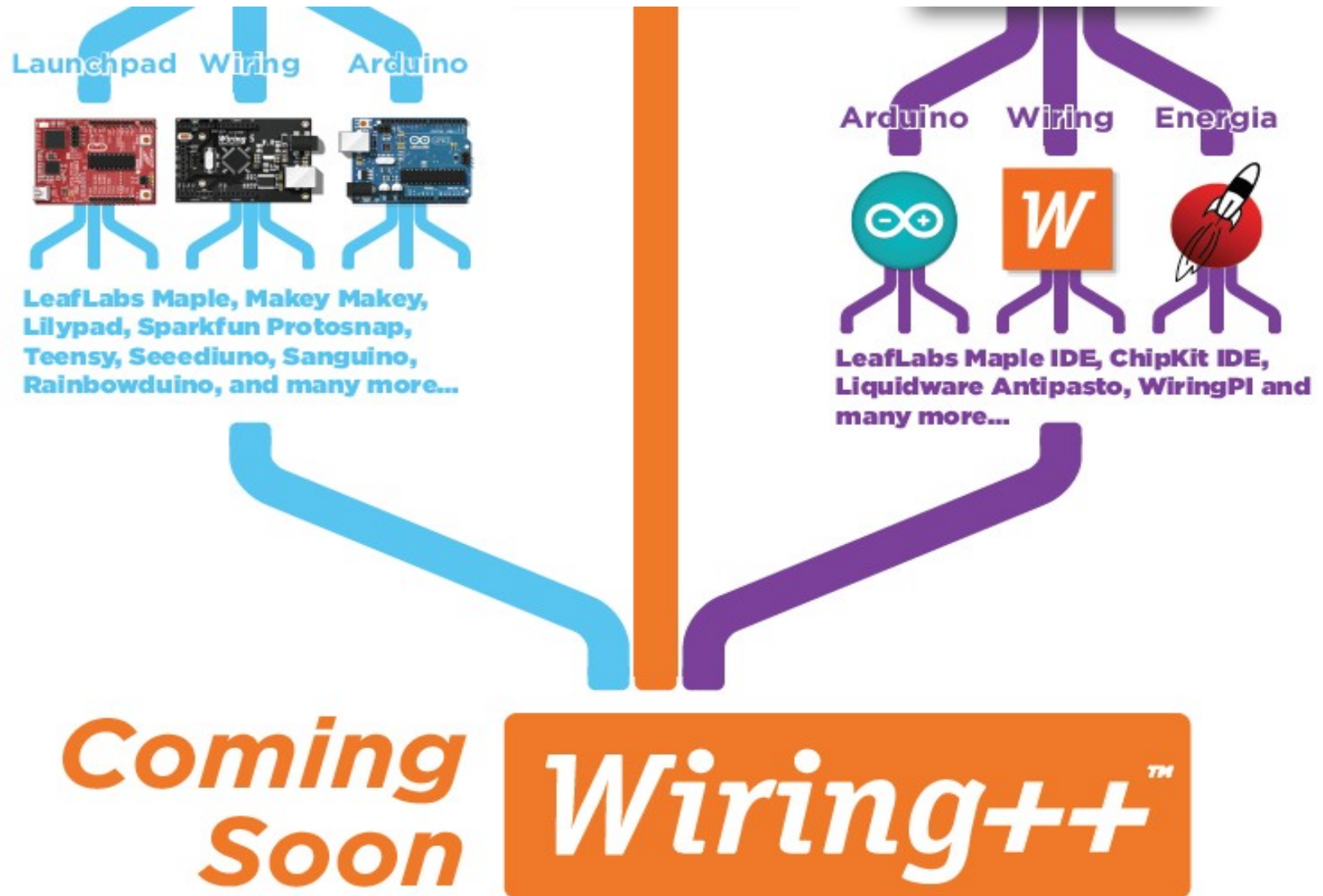
# Egy jól hangzó ígéret

- Ez volt a múlt, de most mindezt...



# Egy jól hangzó ígéret

- ...egyesítjük egy új projektben (Kickstarter kampány 2013-ból)



# Új kunsztok a Wiring++-ban

## Events

```
void events(){  
  when(button.PRESSED){  
    led.toggle();  
  }  
}
```



Program electronics that respond *when* things happen in your environment.

## Threading

```
void threads(){  
  thread(ECHO){  
    while(1){  
      if (Serial.available() > 0) {  
        Serial.write(Serial.read());  
      }  
      delay(100);  
    }  
  }  
}
```



Organize your code in *threads* to easily do more than just one thing at a time.

# Új kunsztok a Wiring++-ban

## Pins as Objects

```
GpioPin led = PIN13;  
led.pinMode(OUTPUT);  
led.digitalWrite(HIGH);
```

++

```
PwmPin fan = PIN12;  
fan.pinMode(OUTPUT).analogWrite(50);
```

Make your code more readable by *clarifying* each pin's functionality.

Chain method calls together for shorter code.

## Dual C & C++ APIs

```
PIN13.digitalWrite(HIGH);
```

++

```
digitalWrite(13, HIGH);
```

Use the C++ API to explore *object oriented* programming.

The C API provides backwards compatibility with Wiring 1.0

# Ami eddig megvalósult...

---

## [Sketching13] Wiring++

Ed Baafi [ed at modk.it](mailto:ed@modk.it)

*Sun Jul 21 15:45:23 PDT 2013*

- Previous message: [\[Sketching13\] konashi.js](#)
- Next message: [\[Sketching13\] Post-Sketching Update #1](#)
- Messages sorted by: [\[date\]](#) [\[thread\]](#) [\[subject\]](#) [\[author\]](#)

---

Hi Guys,

If you want to try threading and events for Wiring/Arduino I've put up an early build that works with WiringS/Uno/Duemilanove etc. There are a bunch of examples under `help->examples->Wiring++`. I've also attached a high-level poster, the in-progress specification, and Hernando Barragan's original Wiring thesis. I encourage you all to read Hernando's thesis as it may address many of your questions about why certain decisions/tradeoffs were made in what many of you refer to as the Arduino language. We look forward to your feedback, questions, and concerns.

[https://s3.amazonaws.com/modkit/prerelease/wiring%2B%2B/Wiring\\_Windows\\_proof\\_of\\_concept\\_Arduino.zip](https://s3.amazonaws.com/modkit/prerelease/wiring%2B%2B/Wiring_Windows_proof_of_concept_Arduino.zip)

[https://s3.amazonaws.com/modkit/prerelease/wiring%2B%2B/Wiring\\_OSX\\_proof\\_of\\_concept\\_Arduino.zip](https://s3.amazonaws.com/modkit/prerelease/wiring%2B%2B/Wiring_OSX_proof_of_concept_Arduino.zip)

Best,  
Ed

# BroadcastFromThreads.pde

```
// Author: Edward Baafi
// (c) 2013 - Modkit LLC

#include "Hardware.h"
EventList{ONESEC,TWOSEC};

void threads(){

  thread(THREAD1){
    while(1){
      //broadcast ONESEC
      //every second
      broadcast(ONESEC);
      delay(1000);
    }
  }

  thread(THREAD2){
    while(1){
      //broadcast TWOSEC
      //every two seconds
      broadcast(TWOSEC);
      delay(2000);
    }
  }
}

void events(){

  //Handle OneSEC Event
  when(ONESEC){
    Serial.println("ONE SECOND EVENT CALLED..");
  }

  //Handle TWOSEC Event
  when(TWOSEC){
    Serial.println("TWO SECOND EVENT CALLED..");
  }
}

// setup is called before any user threads
void setup(){
  Serial.begin(9600);
  THREAD1.start();
  THREAD2.start();
}

//backwards compatible loop
void loop(){
}
```