

Introducing Arduino & Dino: Ruby meets the physical world!

David Grayson
Las Vegas Ruby Group, 2013-1-30

Arduino

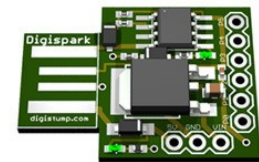
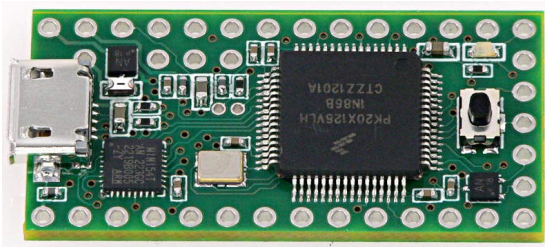
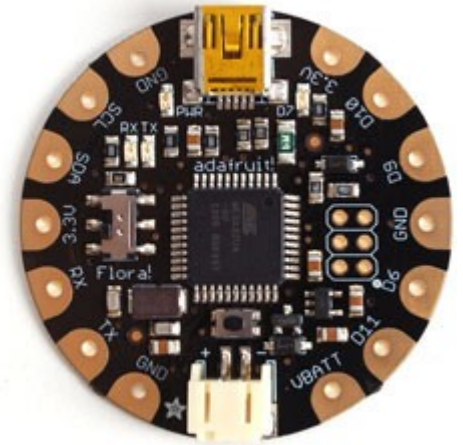
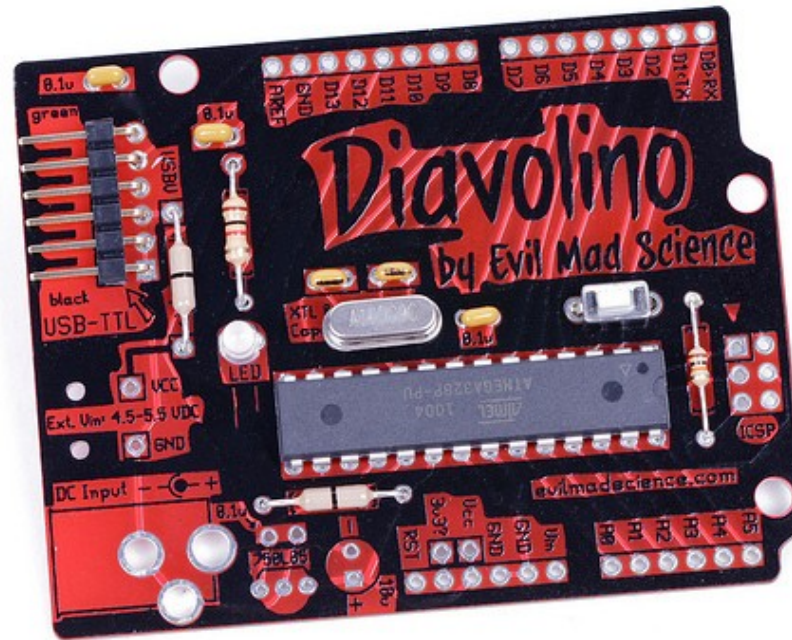
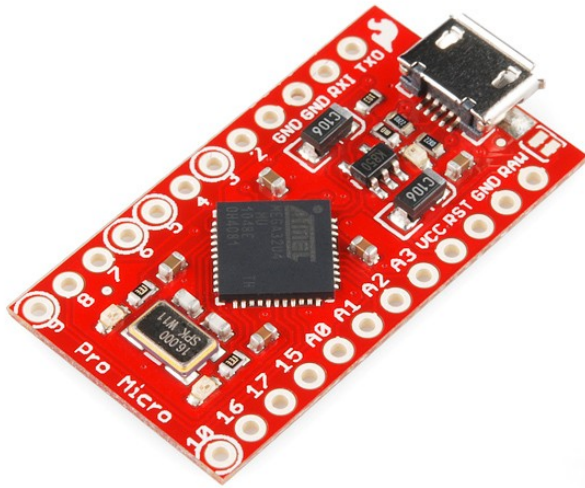


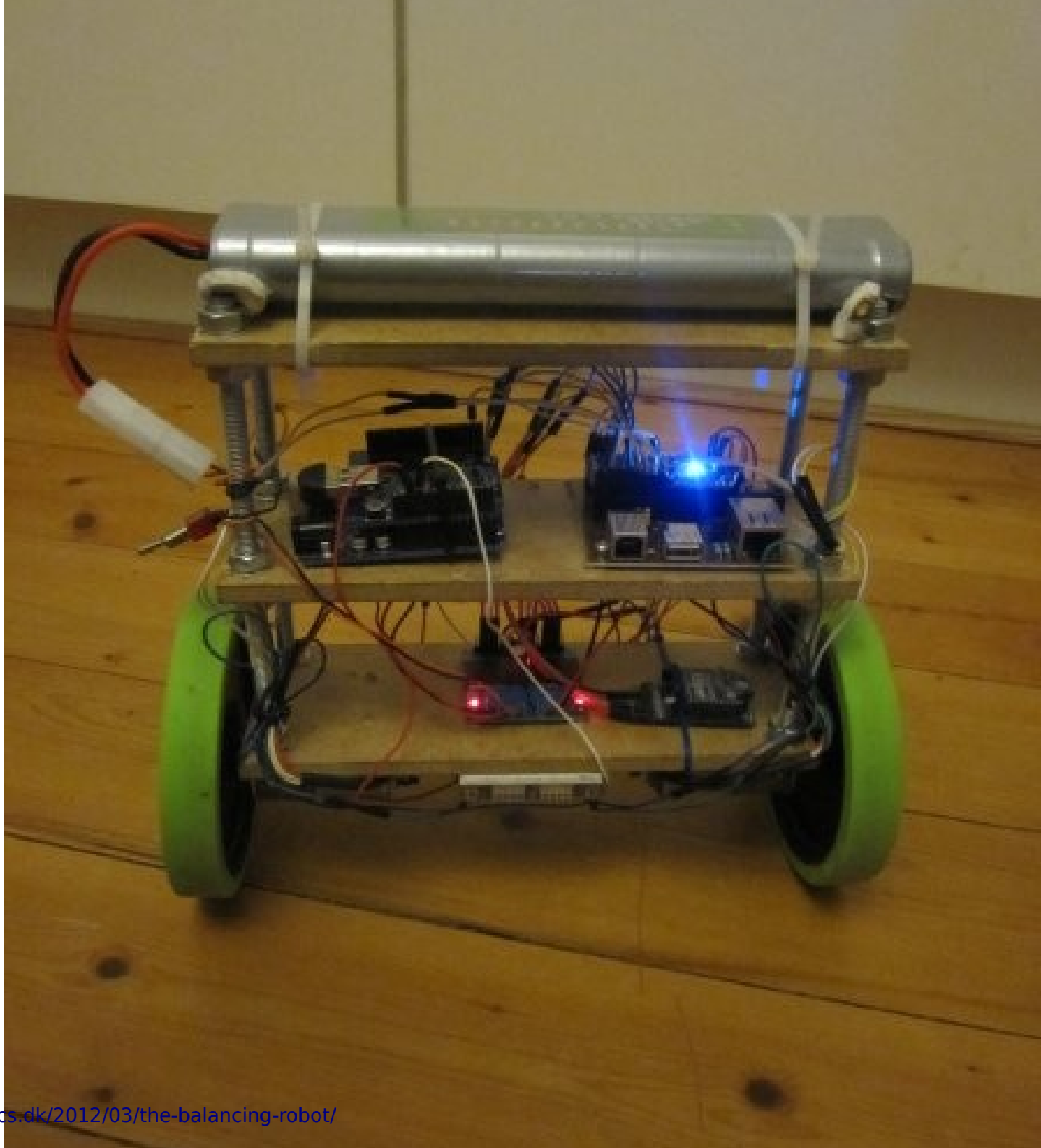
www.pololu.com

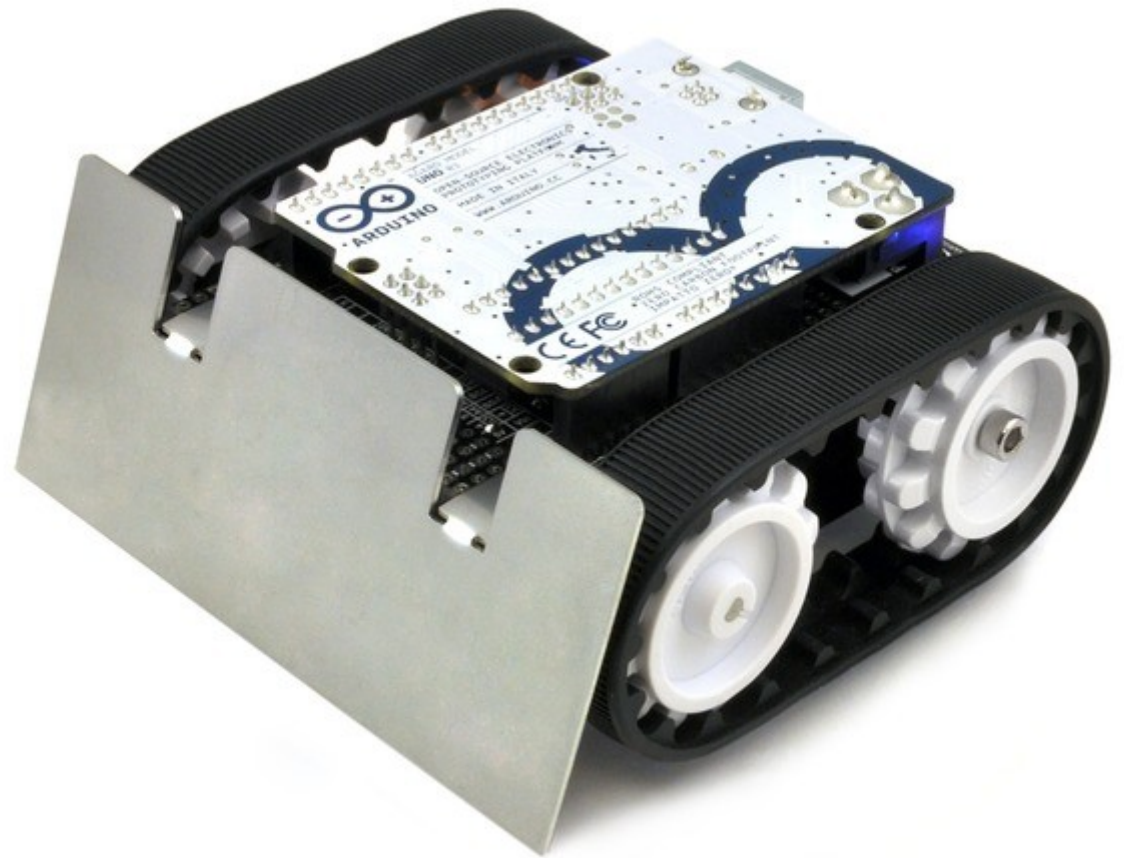
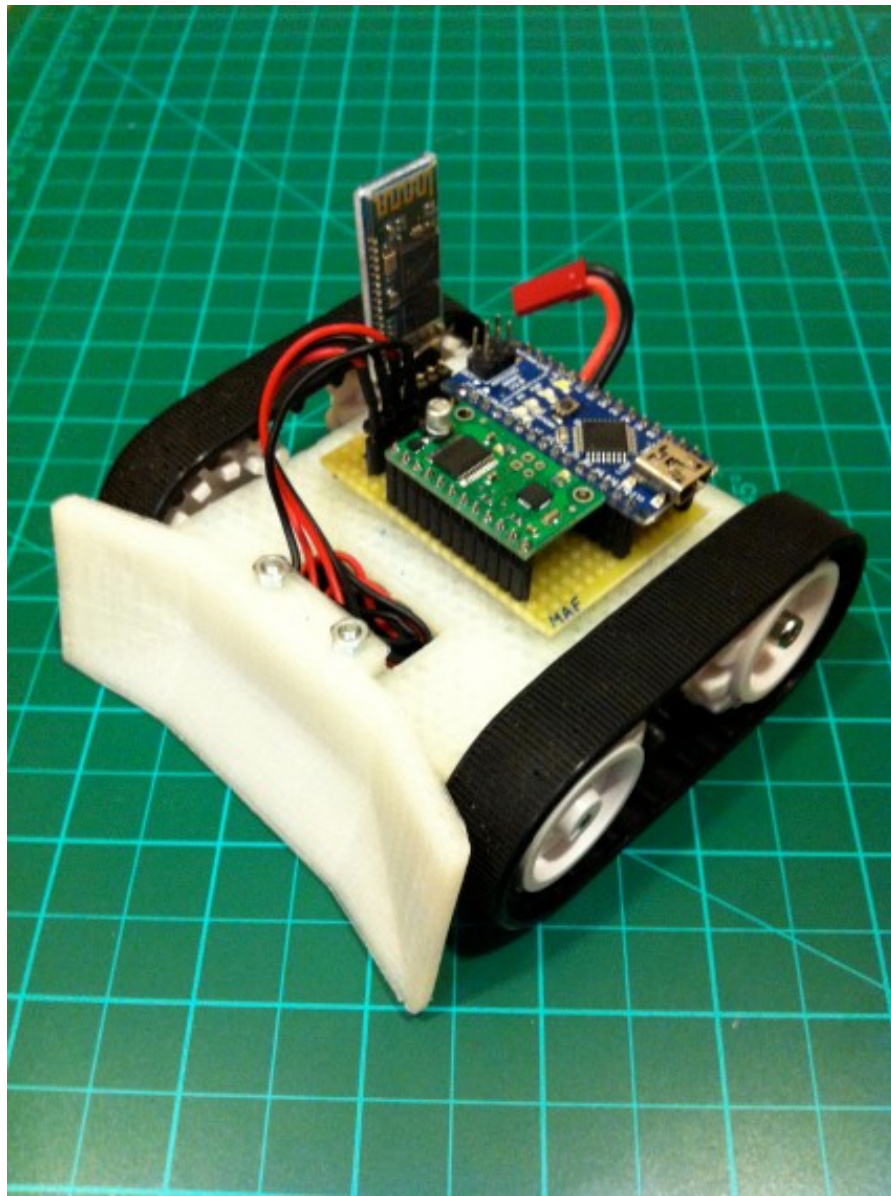


www.pololu.com

Pictures taken from pololu.com, arduino.cc, sparkfun.com







RoboBrrd



So many projects...

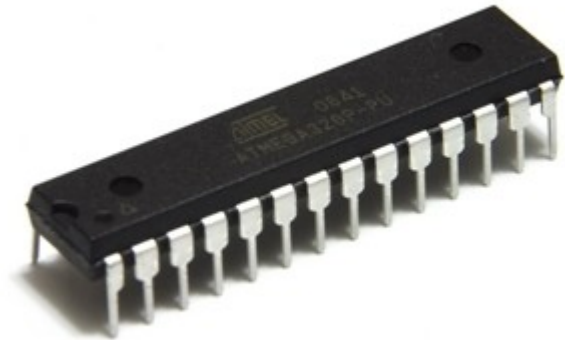
<http://www.instructables.com/id/Arduino-Projects/>

<http://www.pololu.com/resources/communityprojects>

Arduino = AVR + USB + IDE

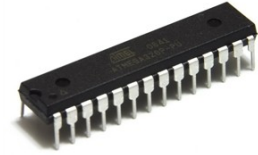


Your PC vs. ATmega328P

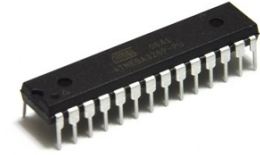




\$1950.00



\$3.16
or less

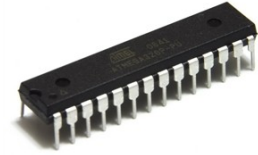


1,006 GB

32 KB



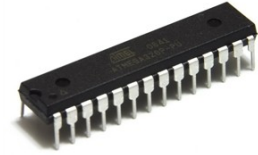
16 GB



2 KB



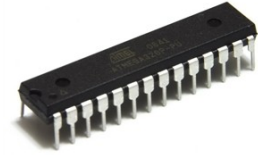
2.4 GHz



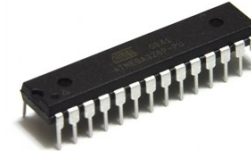
20 MHz



30
seconds

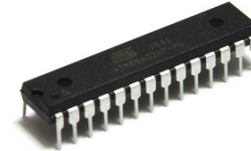


0.003
seconds



Screen,
keyboard,
touchpad,
USB,
ethernet,
HDMI, VGA,
SD card, DVD
drive, WiFi,
Bluetooth

(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	7	22	GND
GND	8	21	AREF
(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)
(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
(PCINT23/AIN1) PD7	13	16	PB2 (\overline{SS} /OC1B/PCINT2)
(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)



Documentation



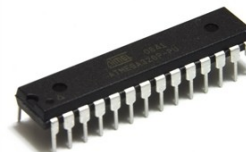
Atmel

Atmel 8-bit Microcontroller with 4/8/16/32KBytes In-System Programmable Flash

ATmega48A; ATmega48PA; ATmega88A; ATmega88PA;
ATmega168A; ATmega168PA; ATmega328; ATmega328P

Features

- High Performance, Low Power Atmel® AVR® 8-Bit Microcontroller Family
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 20 MIPS Throughput at 20MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 4/8/16/32KBytes of In-System Self-Programmable Flash program memory
 - 256/512/512/1KBytes EEPROM
 - 512/1K/1K/2KBytes Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- Atmel® QTouch® library support
 - Capacitive touch buttons, sliders and wheels
 - QTouch and QMatrix® acquisition
 - Up to 64 sense channels
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Six PWM Channels
 - 8-channel 10-bit ADC in TQFP and QFN/MLF package
 - Temperature Measurement
 - 6-channel 10-bit ADC in PDIP Package
 - Temperature Measurement
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Byte-oriented 2-wire Serial Interface (Philips I²C compatible)
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 23 Programmable I/O Lines



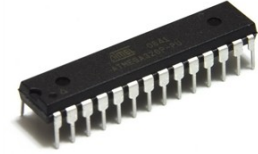
LVBots

LAS VEGAS
RUBY
GROUP



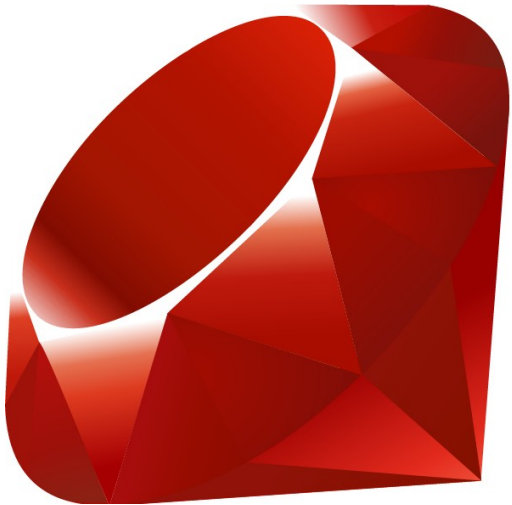
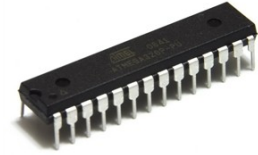


BIOS,
OS,
Processes,
Threads,

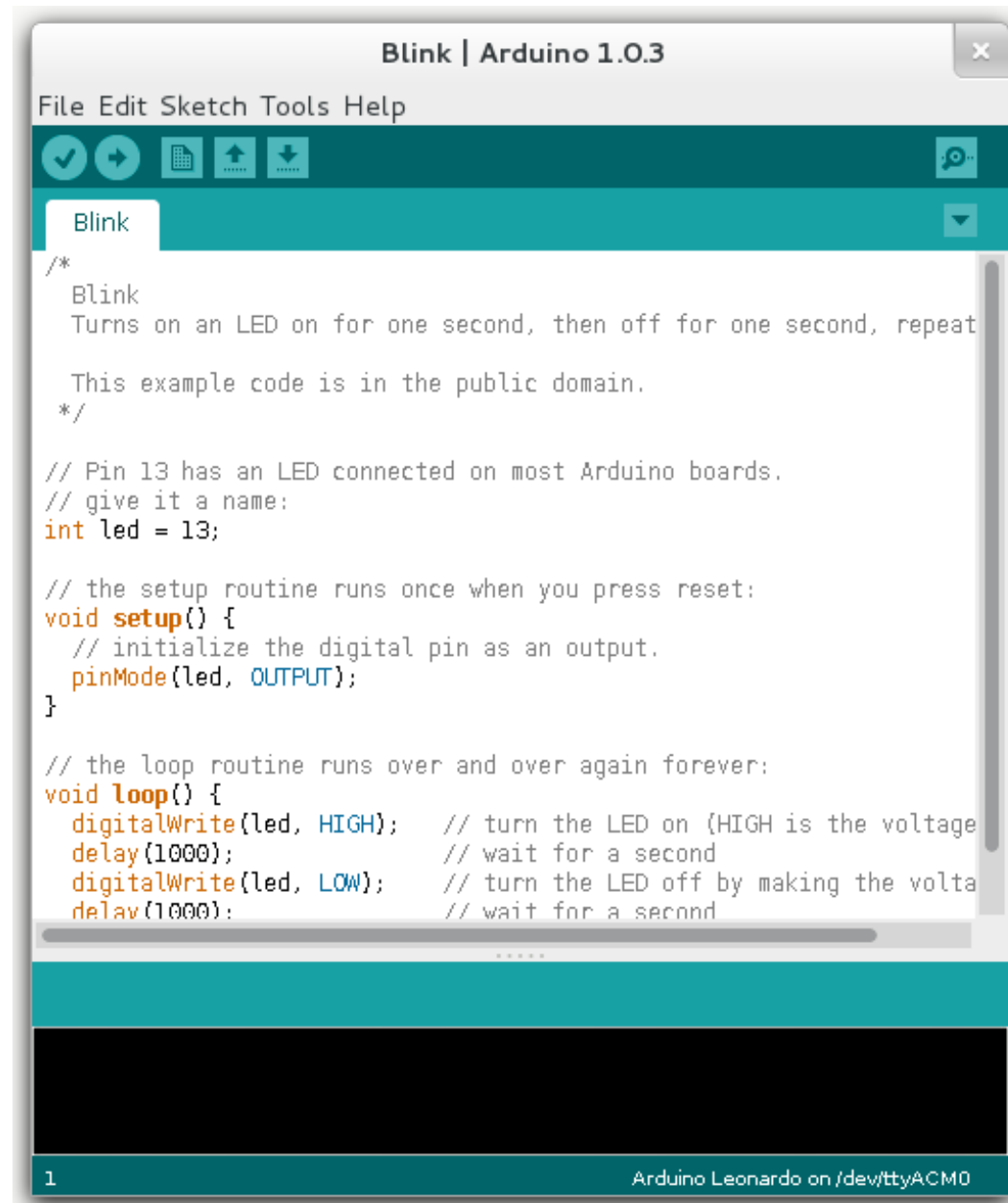


Registers,
Interrupts

(maybe bootloader)

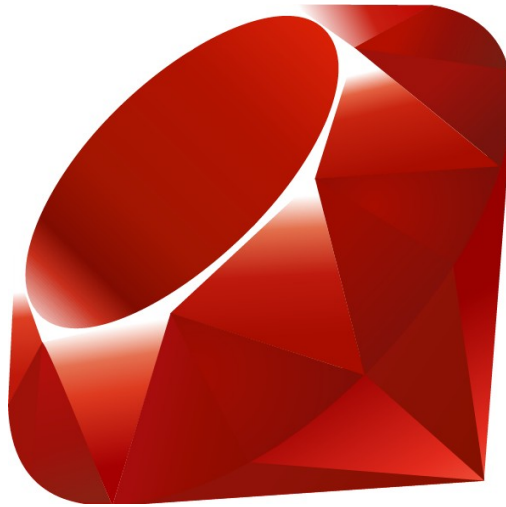


C/C++
avr-gcc



Dino

Control your Arduino from Ruby!



Dino

- Author: Austin Vance
- Source: <http://github.com/austinbv/dino>
- RubyConf 2012 Talk

Blink LED

```
require 'dino'

board = Dino::Board.new(Dino::TxRx.new)
led = Dino::Components::Led.new(pin: 13, board: board)

[:on, :off].cycle do |switch|
  led.send(switch)
  sleep 0.5
end
```

Sensor

```
require 'dino'

board = Dino::Board.new(Dino::TxRx.new)
sensor = Dino::Components::Sensor.new(pin: 'A0',
  board: board)

on_data = Proc.new do |data|
  p data
end

sensor.when_data_received(on_data)

sleep
```

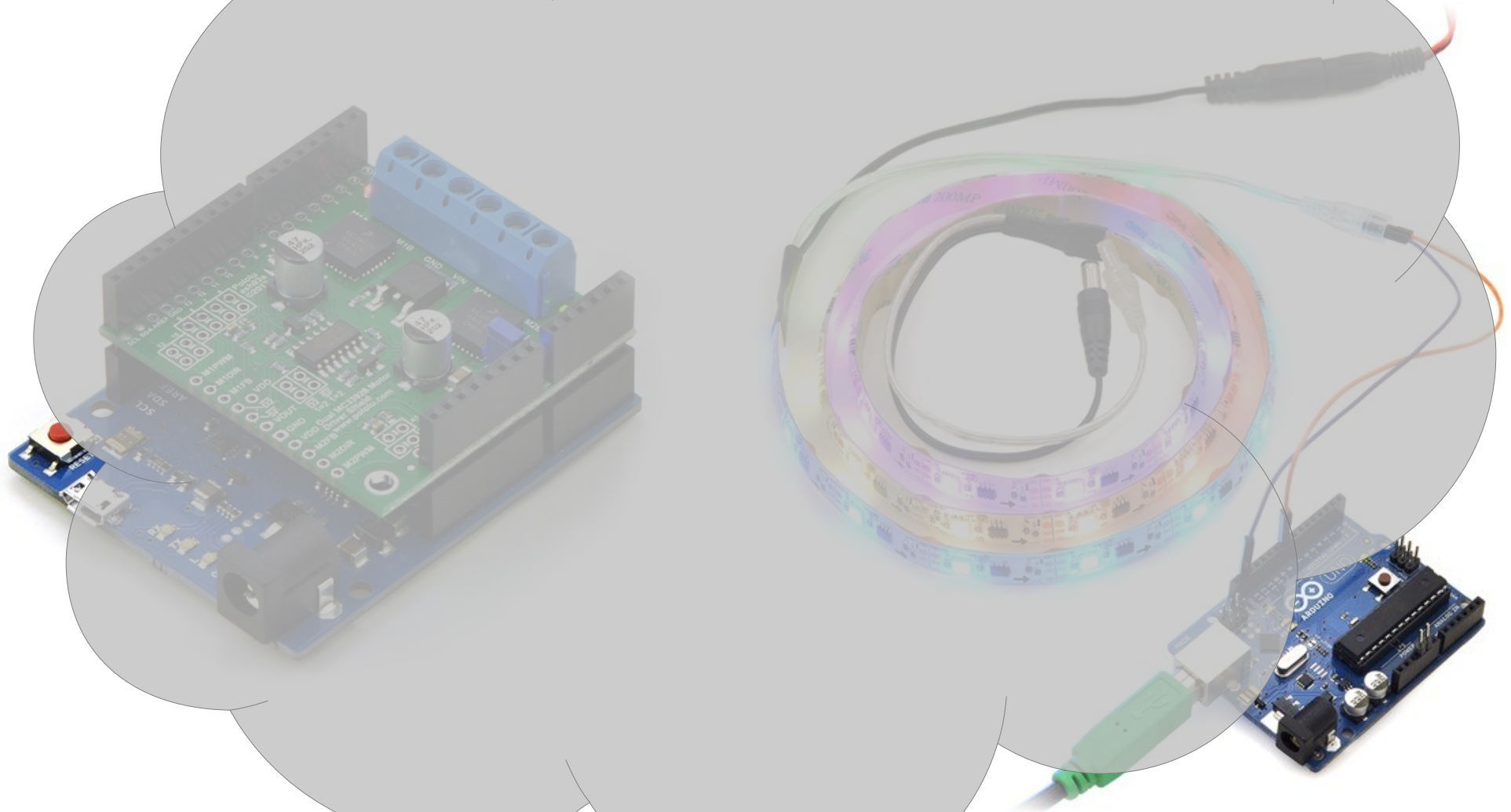
...could be WAY cleaner:

```
require 'dino'
```

```
board = Dino::Board.new  
sensor = board.analog_inputs[0]
```

```
puts sensor.voltage # => 4.0185546875
```


Shields & Libraries



Other components...

Button



IR Receiver

RGB Led



Servo



Stepper



www.pololu.com

More info about Dino:

<http://playground.arduino.cc/interfacing/ruby>

<https://github.com/austinbv/dino>

<http://confreaks.com/videos/1294-rubyconf2012-arduino-the-ruby-way>

Las Vegas Mini Maker Faire®

Saturday February 2nd

Tickets now available on
makerfairevegas.com

FAMILY
FRIENDLY
FUN!

Cirque Mechanic

SYN Shop

Twilio

Pololu

Romotive

Owl Posse

Hydrogadget

Full Spectrum Laser

(and more!)

Historic Fifth Street School, 9am - 6pm

401 S. Fourth St., Las Vegas, NV 89101

A convergence of engineering, crafting, science, education
and all things Maker! The world's greatest show and tell!

PROUDLY SPONSORED BY

Make:
makezine.com

Pololu
Robotics & Electronics

 **SYN SHOP**
THE LAS VEGAS HACKERSPACE

DOWNTOWNPROJECT