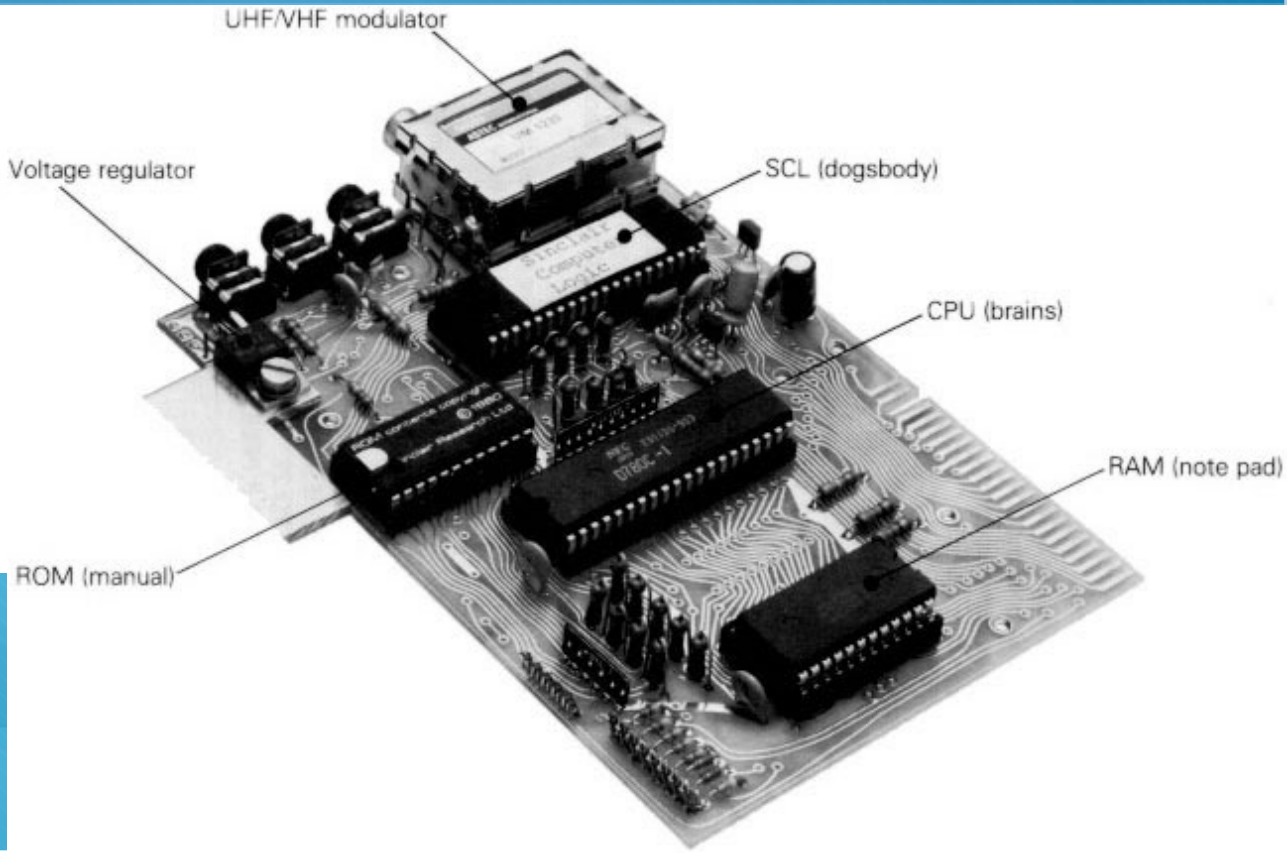
Raspberry pi

Riccardo Barberi, docente di fisica

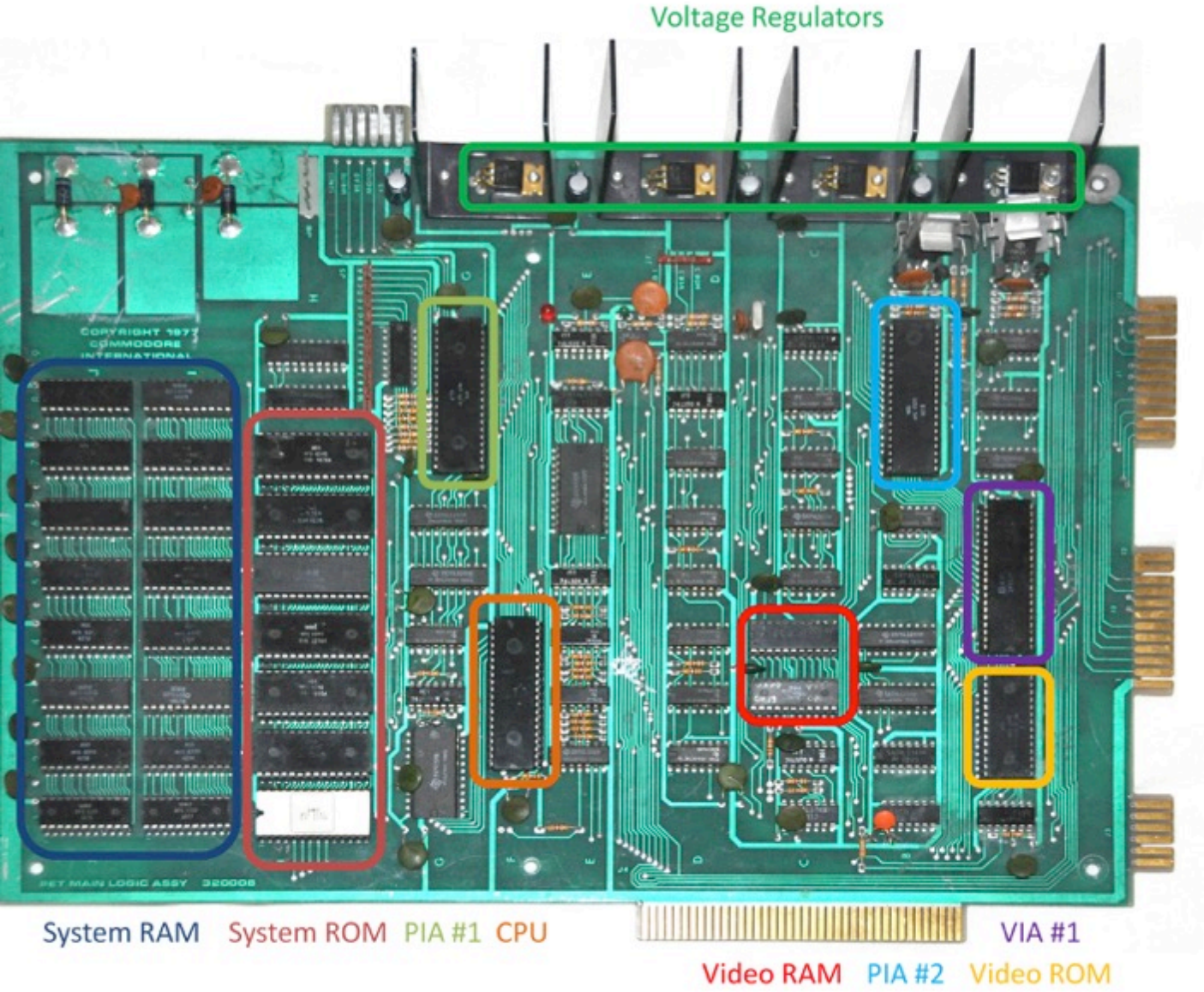
Tanto tempo fa … (anni ‘80)

nei computer potevi guardarci dentro

1980



Commodore PET 2001 Main Board Layout



1977

PIA – peripheral interface adaptor, MOS 6520

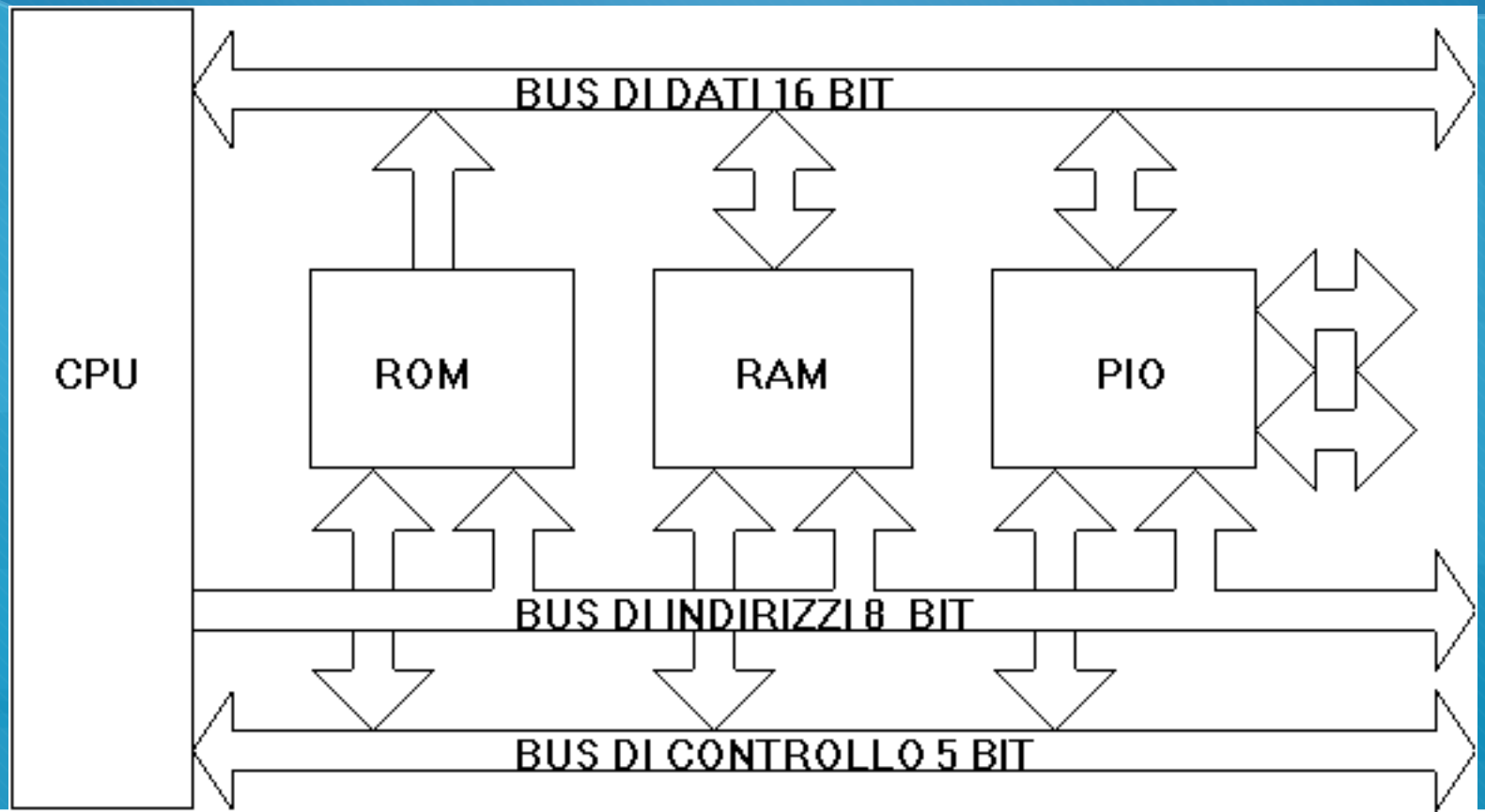
VIA - Versatile interface adapter, MOS 6522

ROM – memoria di sola lettura

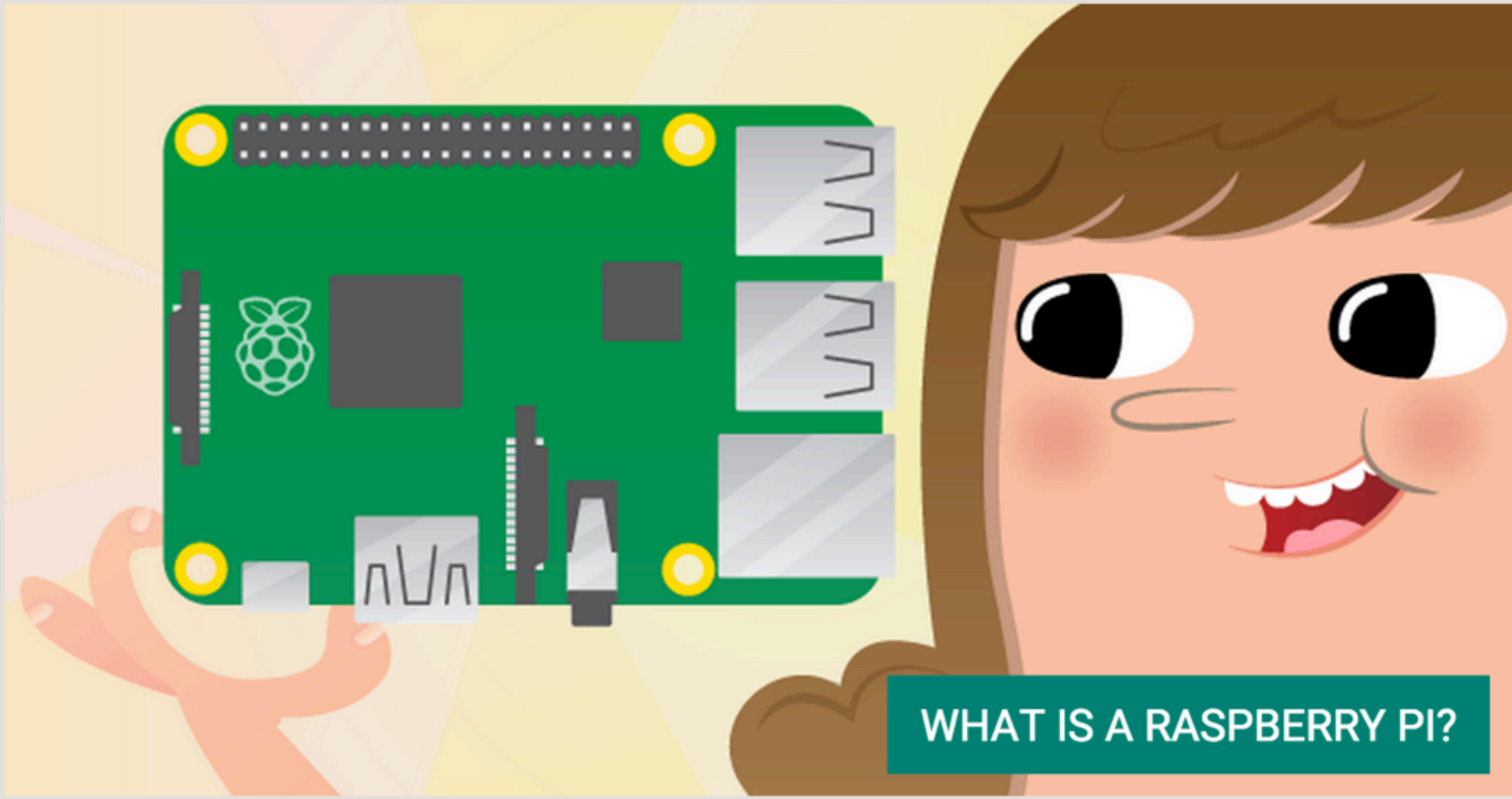
RAM – memoria elettronica di lettura scrittura, memoria centrale, 8 kByte

CPU – Central Processing Unit, Unità di elaborazione centrale, MOS 6502

Schema tipico PC anni ’80

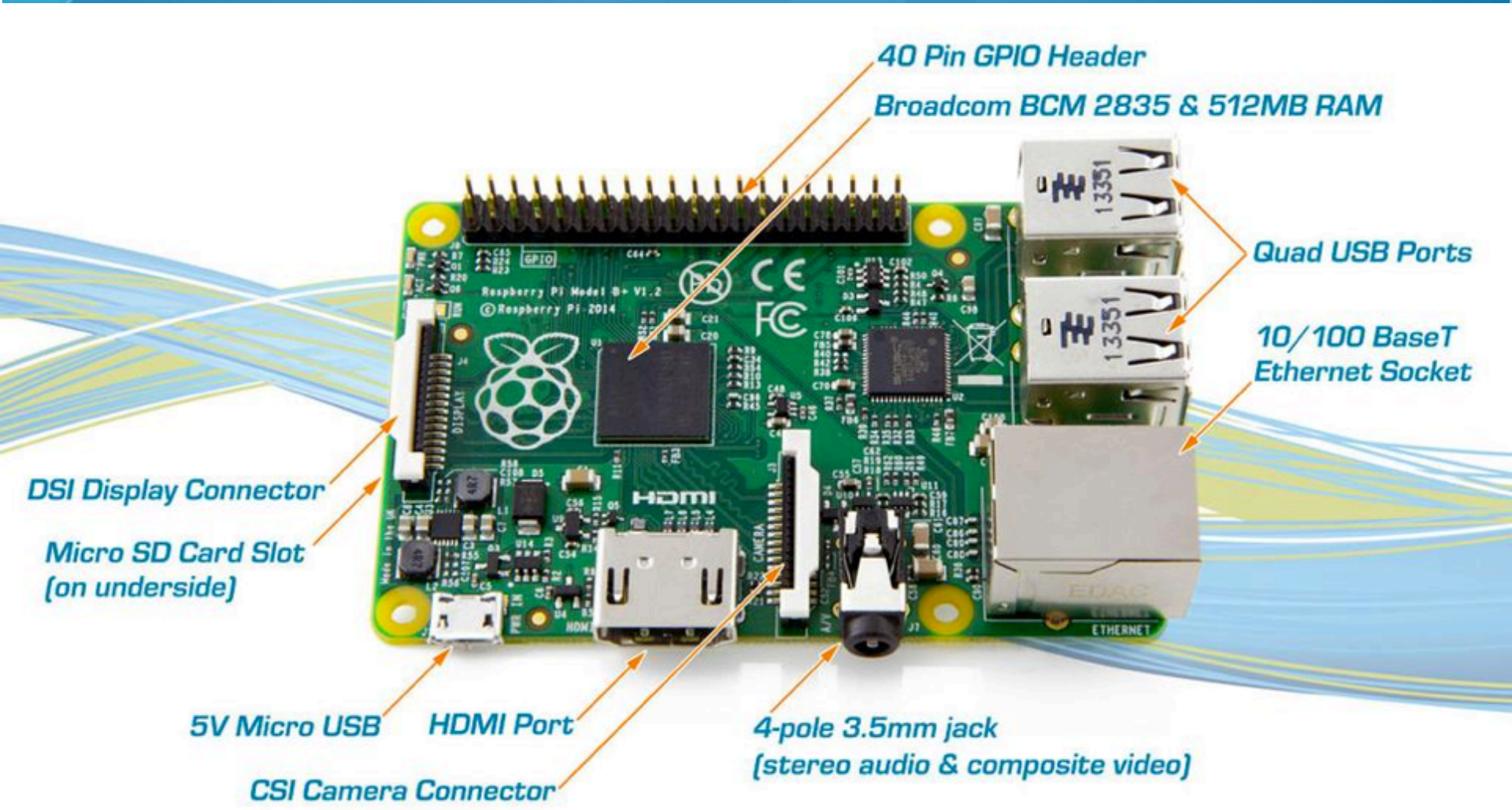


Raspberry pi

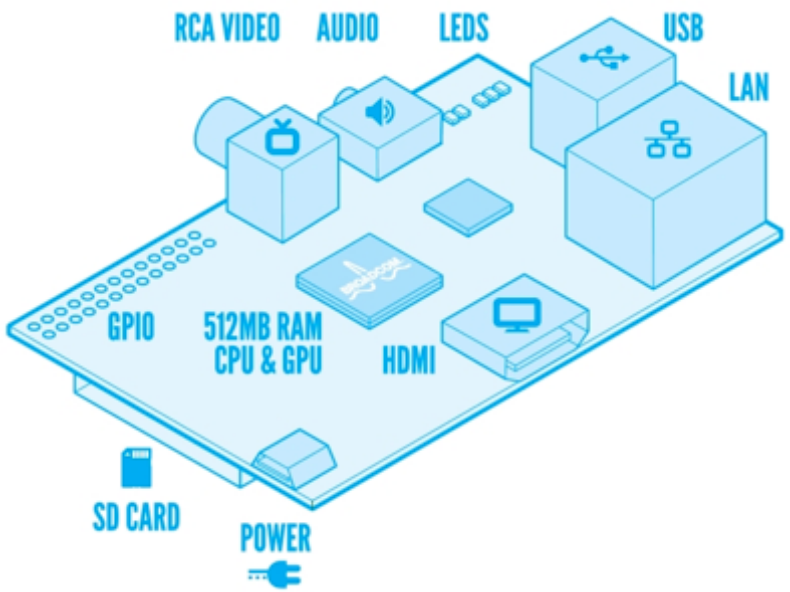


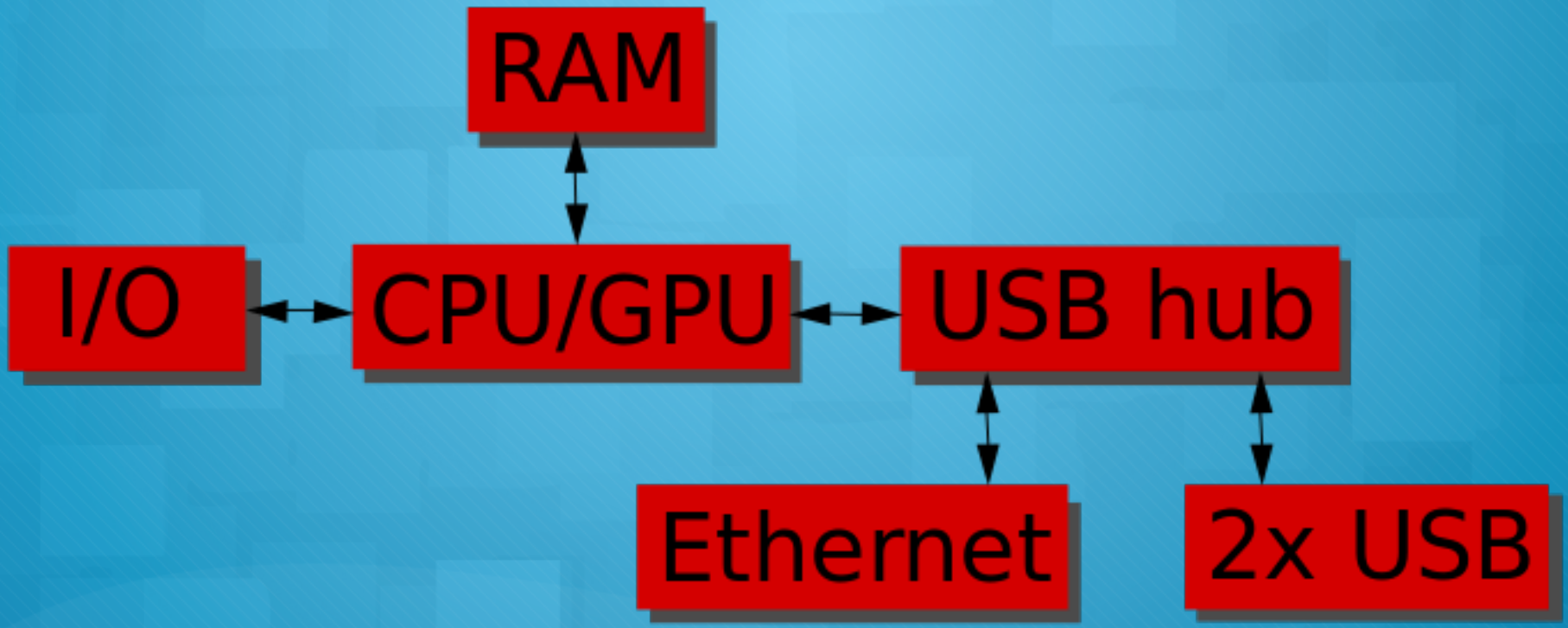
Low cost, credit-card sized computer,
running Linux and other O.S., with the ability to interact with the outside world

La scheda elettronica



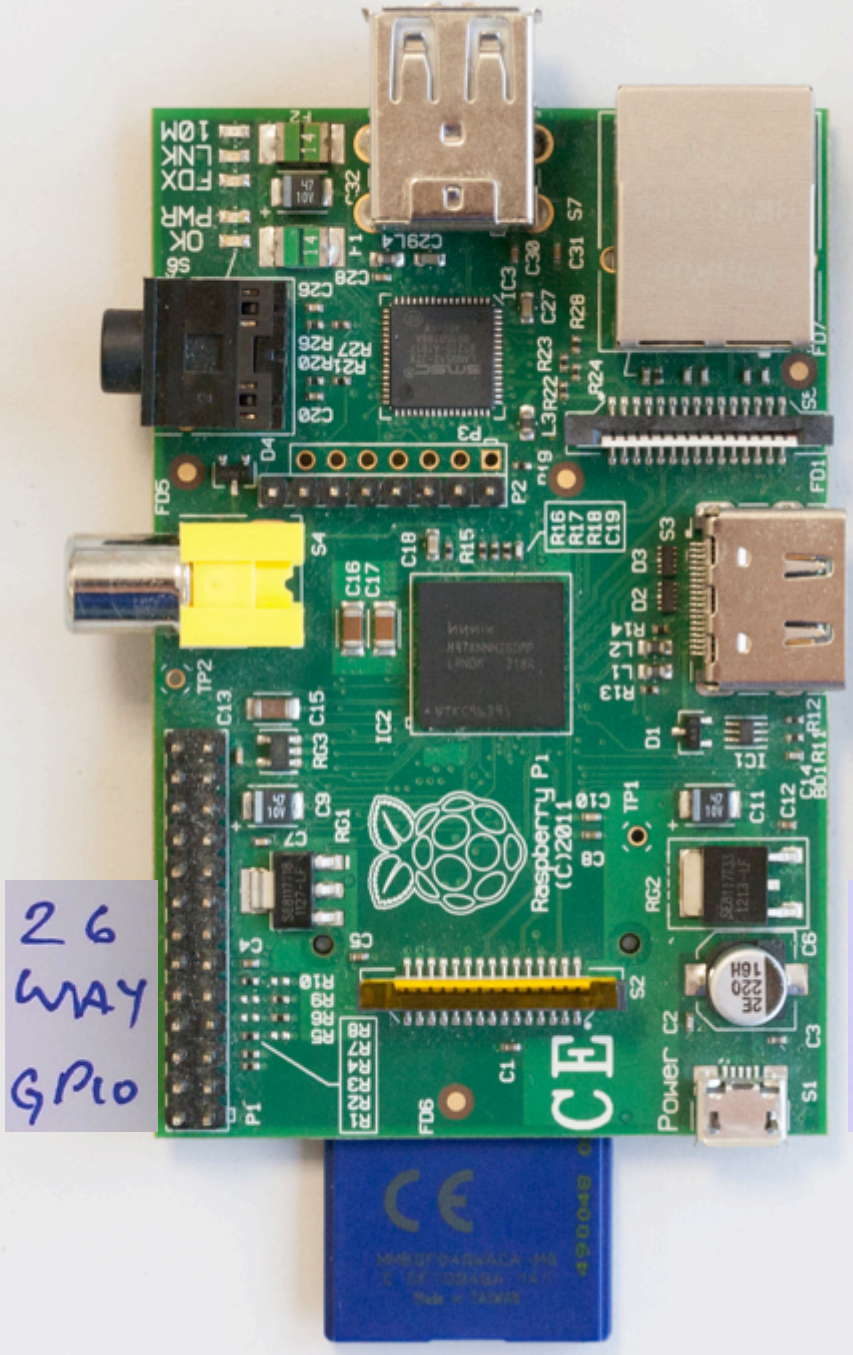
Schema a blocchi del Raspberry Pi

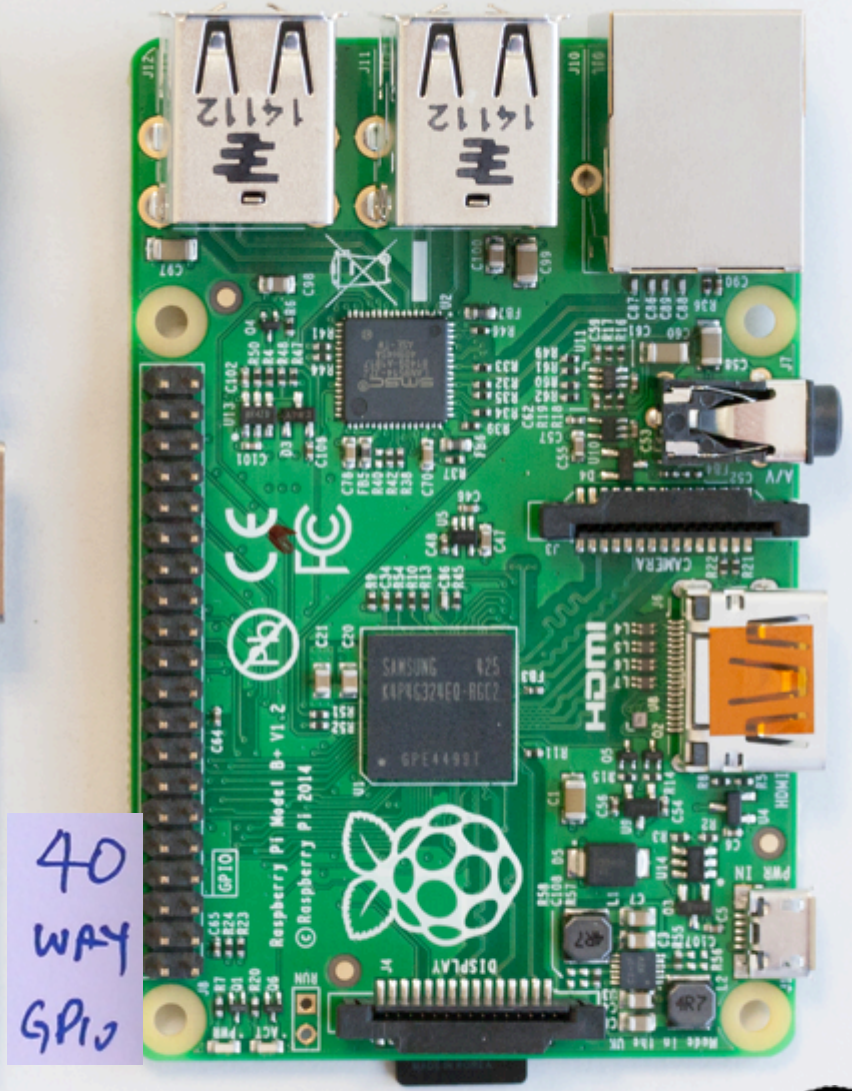




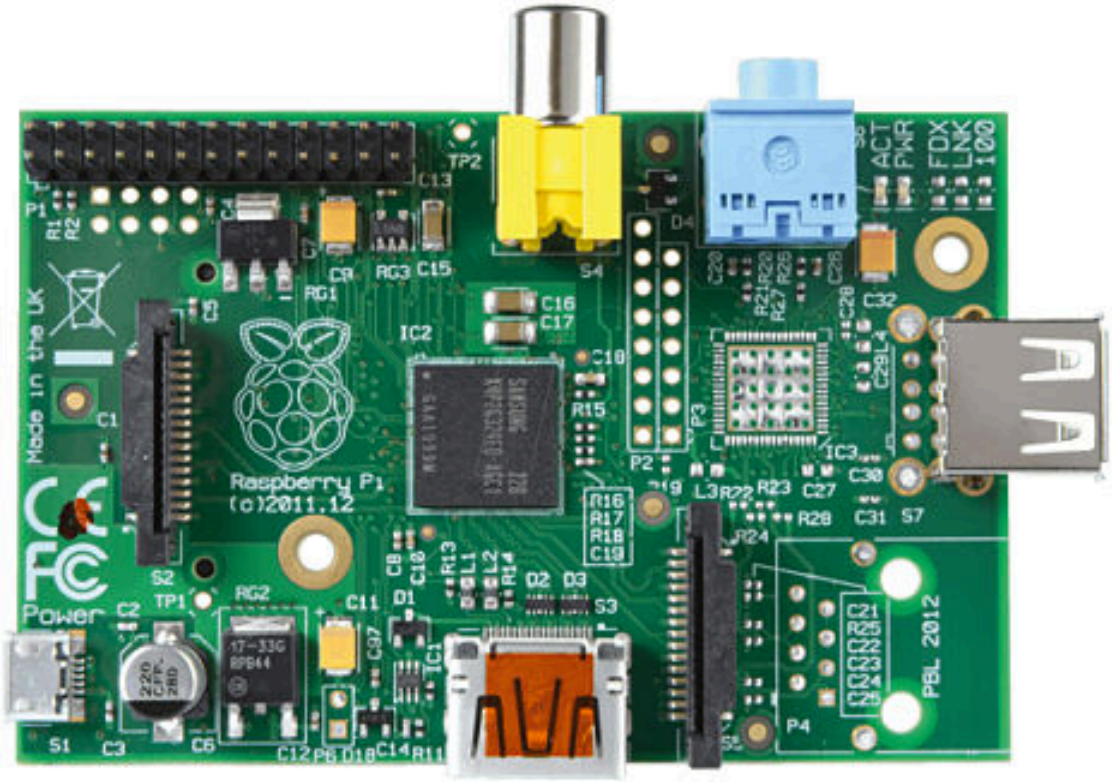
Model B

Model B +

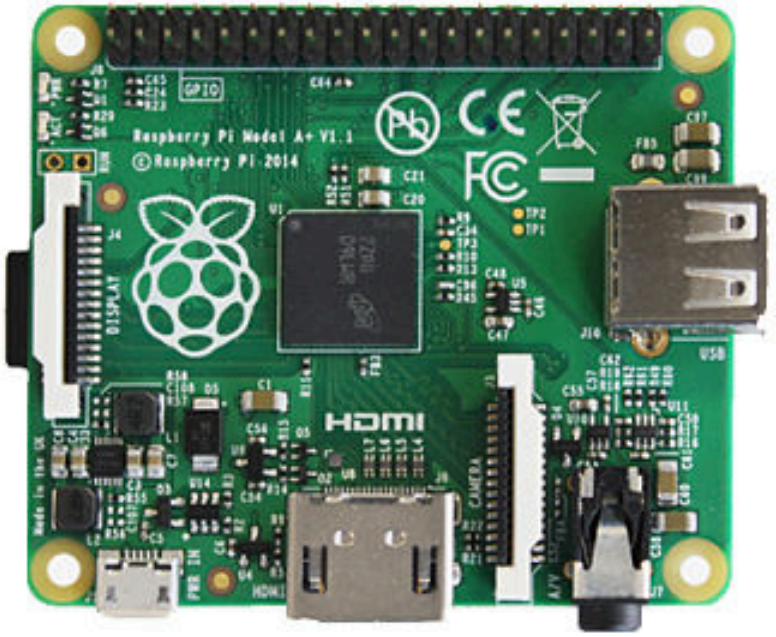




Modelli a basso consumo

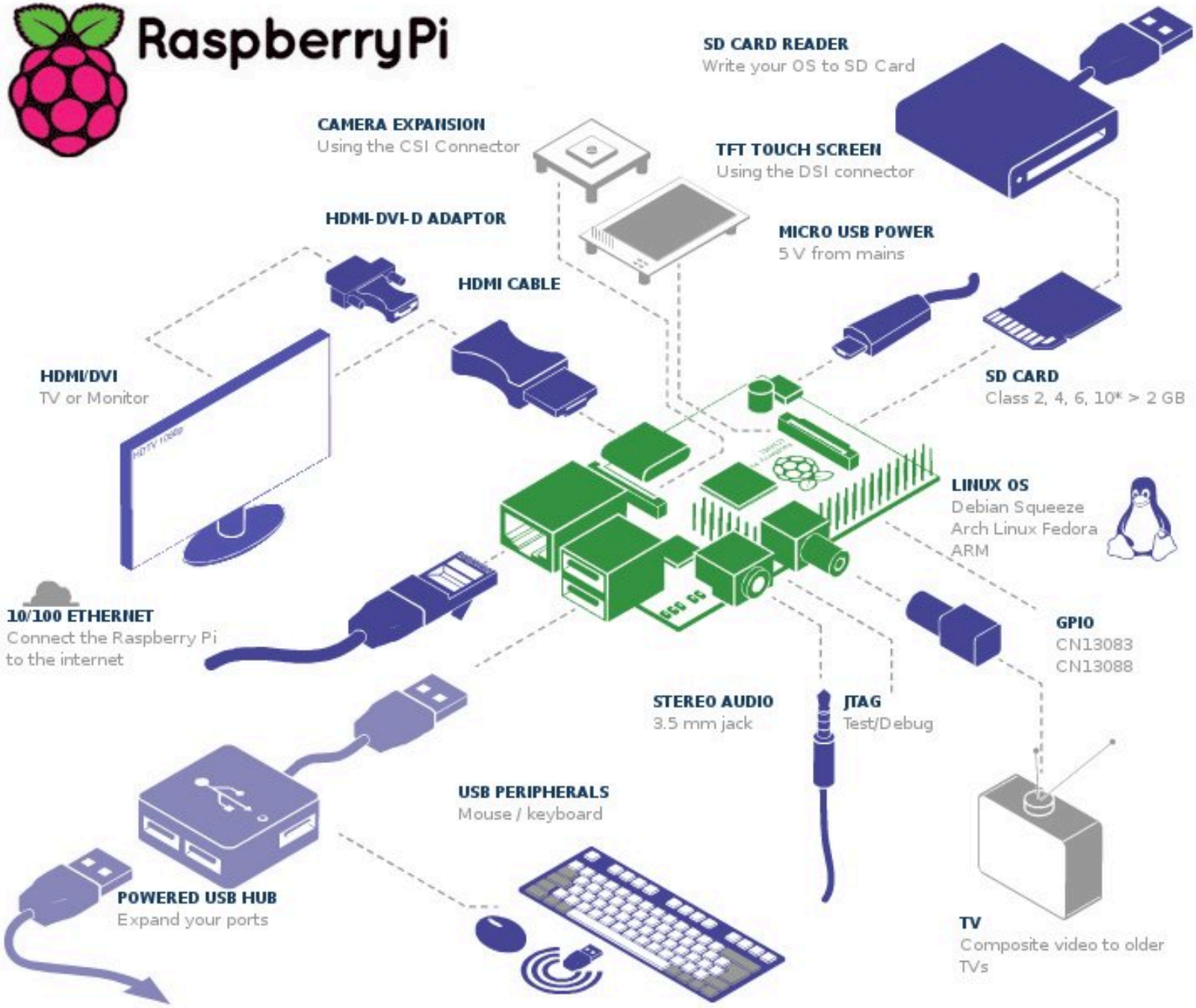


Raspberry Pi Model A



Raspberry Pi Model A+

RaspberryPi

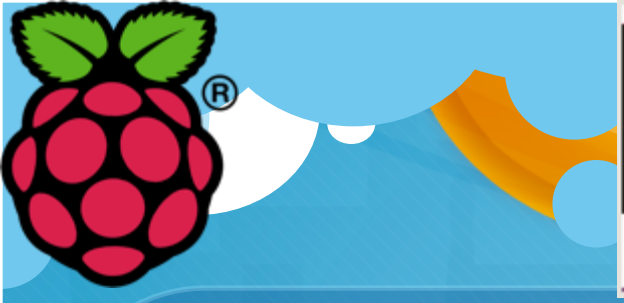


C'è solo il raspberry?

Specs	Raspberry-Pi-2-B	Raspberry-Pi-B+	Odroid-C1	Humming-Board-i2ex	Banana-Pi-PRO
Image					
Released	February 2015	February 2012	December 2014	April 2014	October 2014
CPU SoC	Broadcom BCM2836	Broadcom BCM2835	Amlogic S805	Freescale i.MX6 Dual	Allwinner A20
Chipset	ARM v7 Quad Core	ARM v6 Single Core	ARM v7 Quad Core	Quad Core	ARM v7 Dual Core
Architecture	ARM Cortex-A7	ARM11-76JZF-S	ARM Cortex-A5	ARM Cortex-A9	ARM Cortex-A7
Clock Speed	900MHz	700MHz	1.5GHz	1GHz	1GHz
RAM	1GB	512MB	1GB	1GB	1GB
Memory Type	LP-DDR2	SDRAM	DDR3	DDR3	DDR3
Memory Speed	450MHz	400MHz	?	1066MHz	?
Storage	MicroSD Card	MicroSD Card	MicroSD Card	MicroSD, mSATA	SD Card, SATA 2.0
USB (2.0)	4	4	4	4	2
USB OTG	-	-	1	-	1
Power	1.8A @ 5V / 3W	1.8A @ 5V / 3W	2A @ 5V	2A @ 5V	2A @ 5V
GPIO					
Ethernet Ports					
Ethernet Speed					
GPU					
Video Output					
Audio Output					
Supported OS					
Price					

No! ma il raspberry ha una grande community

https://www.raspberrypi.org/community/



THE MAGPI

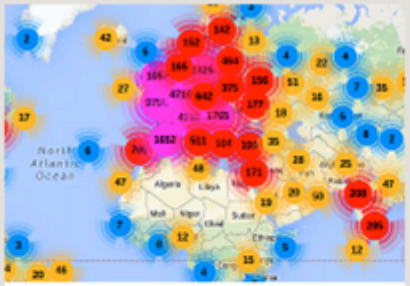
PI WEEKLY


GEEK GURL DIARIES

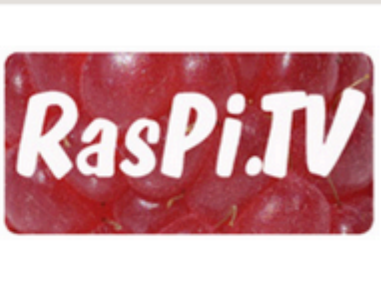
RASPBERRY JAM

Community

RASPBERRY PI TODAY


RASTRACK

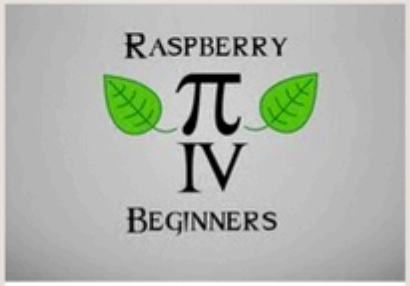

THE RASPBERRY PI GUY



RASPI.TV

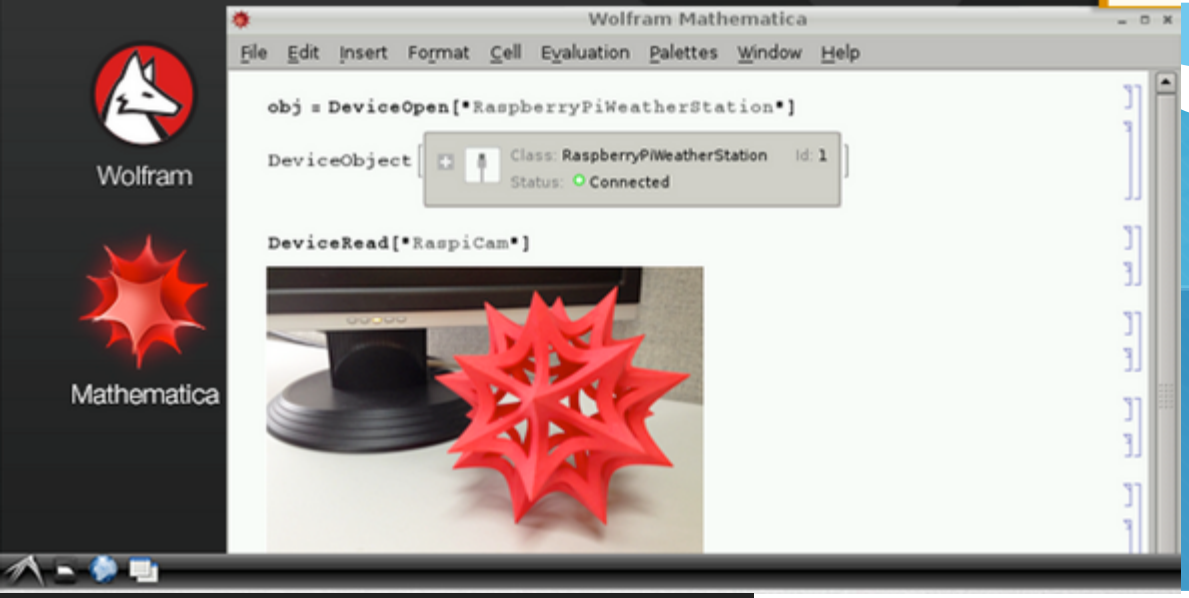
RASPBERRY PI POD


DAVE AKERMAN



RASPBERRY PI IV BEGINNERS

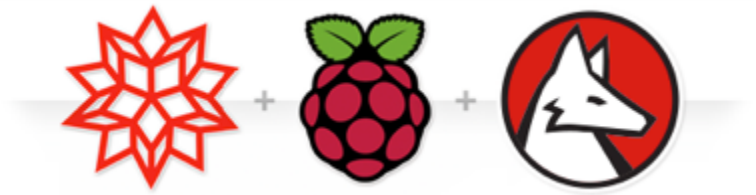

WOLFRAM COMMUNITY



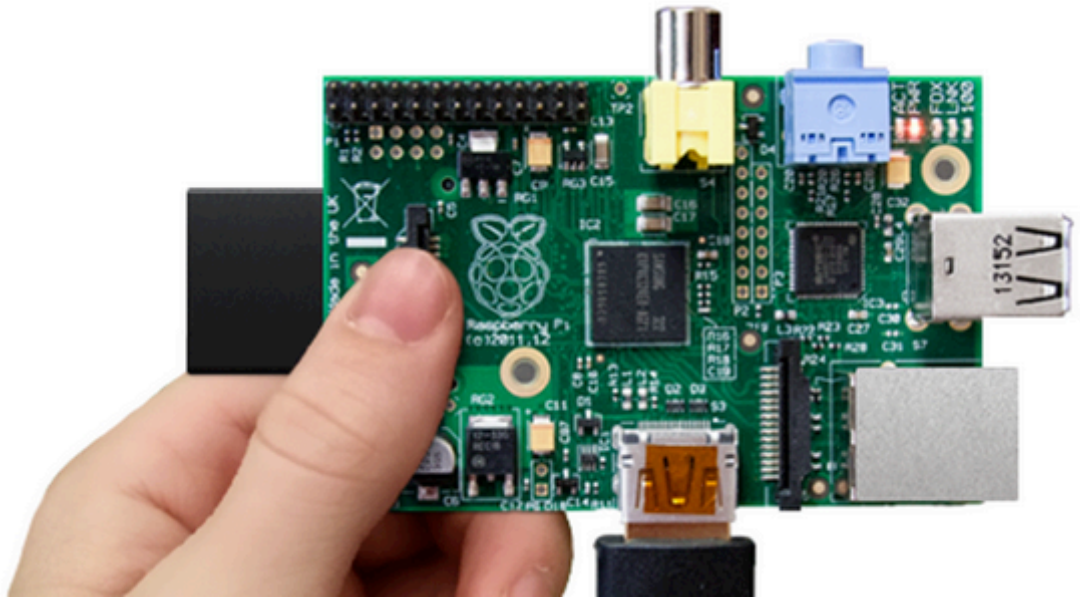
Mathematica gratuito per la seconda volta nella storia, dopo il NeXT di Steve Jobs (1985)

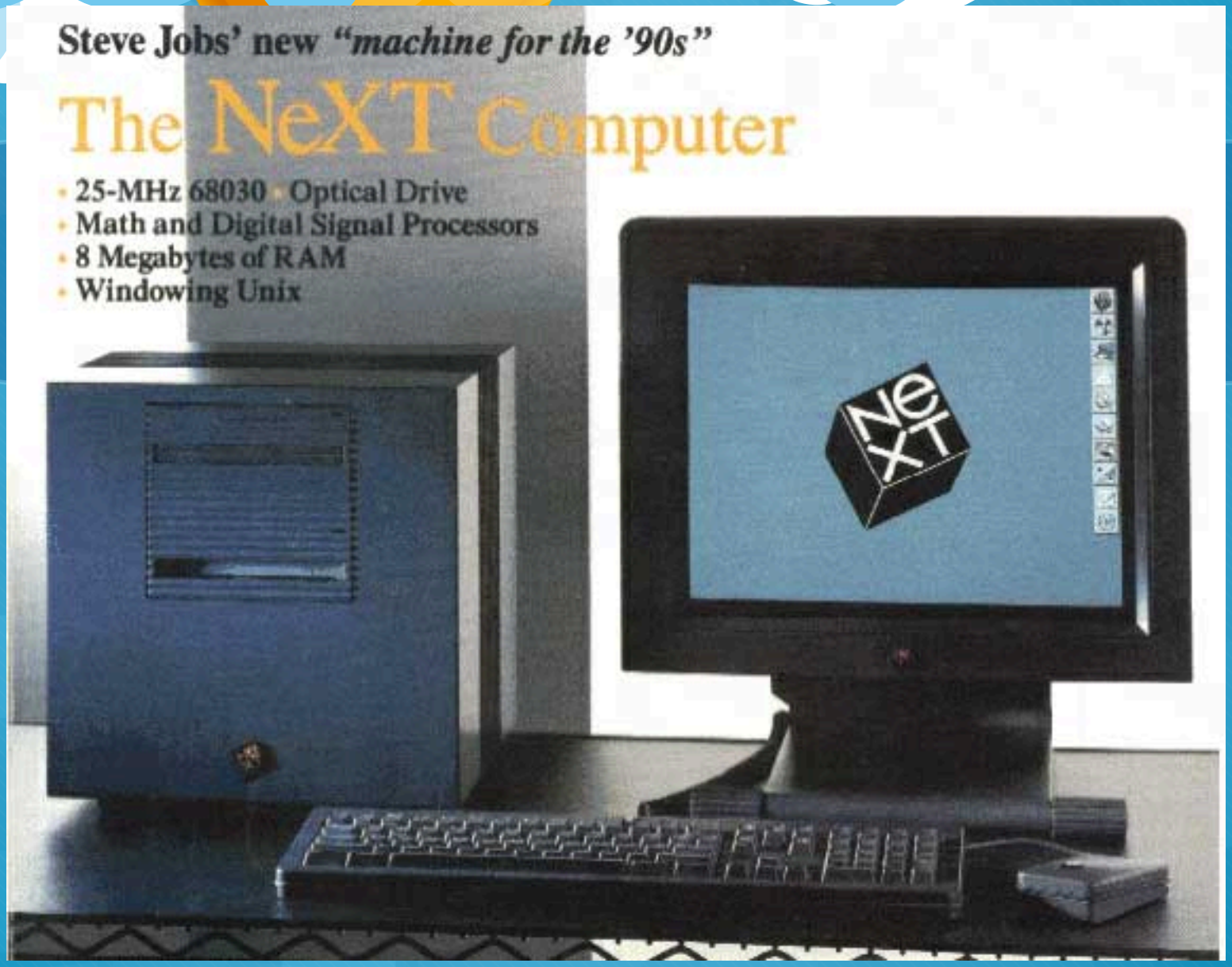
WOLFRAM
COMPUTATION MEETS KNOWLEDGE

Wolfram Language & Mathematica free on every Raspberry Pi!



In alternativa, su un PC o Mac, Mathematica home edition a 295 € + IVA




Steve Jobs' new "machine for the '90s"
The NeXT Computer
• 25-MHz 68030 • Optical Drive
• Math and Digital Signal Processors
• 8 Megabytes of RAM
• Windowing Unix
NeXT

Next PC

- Motorola 6830 – 33 MHz + 68040 math coprocessor
- DSP Chip Motorola 56001
- 8-128 Mbyte RAM
- 660 MB Hard Disk
- Ethernet 10 Mbit
- 20 MIPS, 5.6 MFlops
- 3000 $
- 1985 - 1996

Raspberry pi 2

- Quad core ARM Cortex A7 900 MHz
- GPU Videocore IV
- 1 Gbyte SRAM
- At least 8 Gbyte microSD
- Ethernet 10/100 Mbit
- 2000 MIPS, 93 Mflops (1 core)
- 40 $
- 2015

Cosa serve per usare il raspberry pi?

Stand alone

- Tastiera + mouse
- 1 scheda microSD
- 1 Monitor (pal, secam o hdmi)
- 1 cavo ethernet o 1 adattatore usb-wifi
- Preferite soluzioni di recupero ☺

In rete con un PC/Mac

- 1 scheda microSD
- 1 cavo Ethernet
- Software SSH (gratuito)
- Software VNC (gratuito)

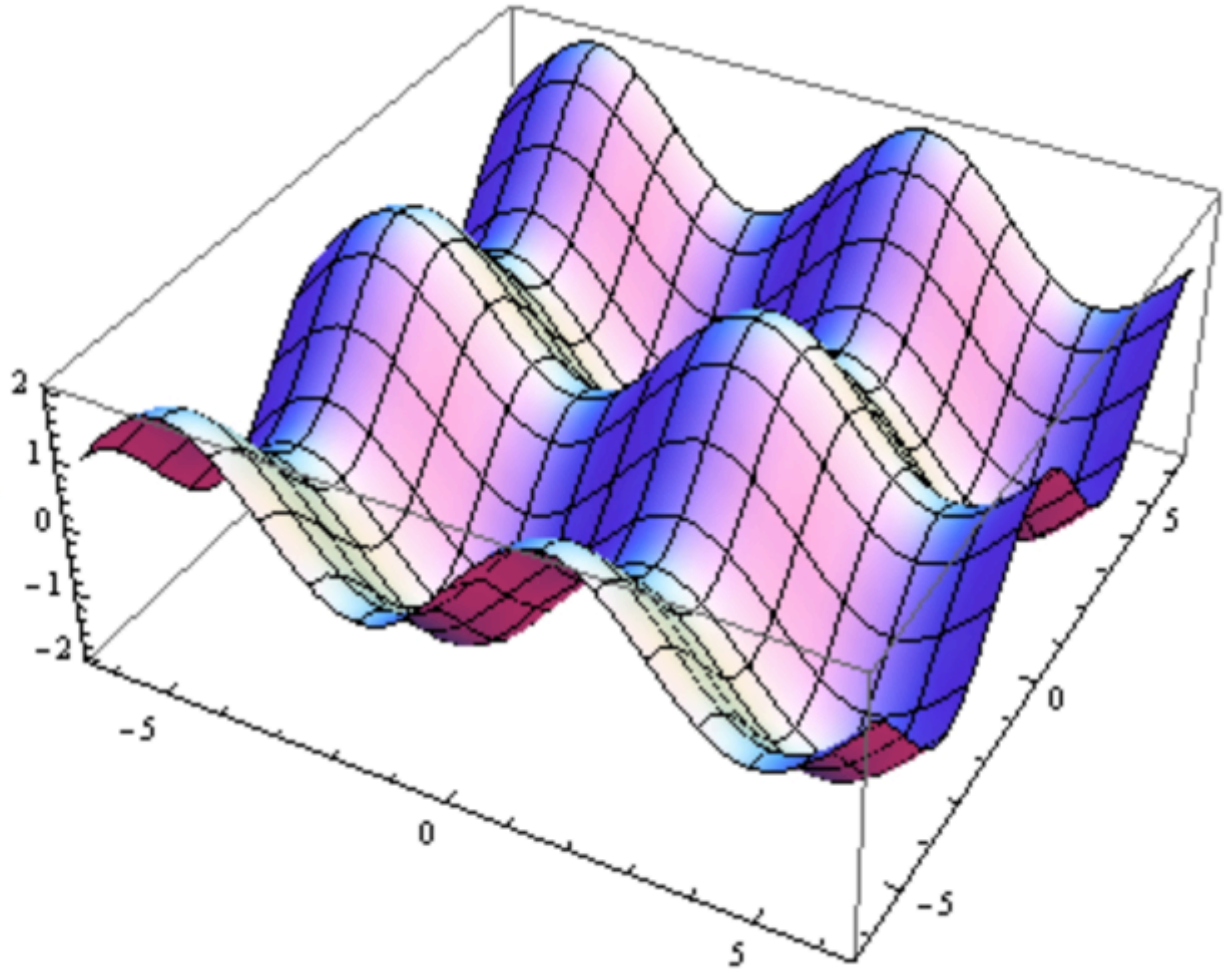
In[1]:=

plot sinx + cosy

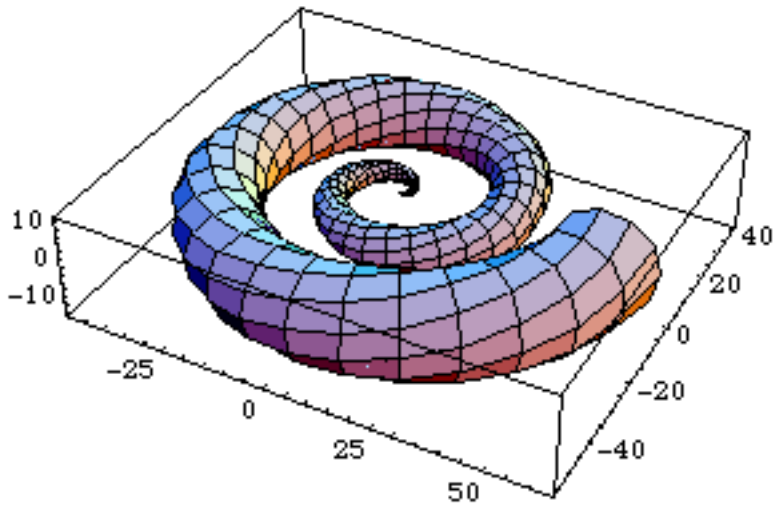
↳ 3D plot

```
Plot3D[Cos[y] + Sin[x], {x, -6.283, 6.283}, {y, -6.283, 6.283}]
```

Out[1]=



```
ParametricPlot3D[{u Cos[u] (4 + Cos[v + u]),
 u Sin[u] (4 + Cos[v + u]), u Sin[v + u]},
{u, 0, 4 Pi}, {v, 0, 2 Pi}, PlotPoints → {60, 12}]
```



- Graphics3D -

Ian-dio1.nb

```
Solve[x^2 + y^2 == 25 /. {x → 3}]
```

$$\{\{y \to -4\}, \{y \to 4\}\}$$

```
Solve[x^2 + y^2 == n^2, {x, y, n}]
```

Solve::svars : Equations may not
give solutions for all "solve" variables. More…

$$\left\{\left\{x \to -\sqrt{n^2 - y^2}\right\}, \left\{x \to \sqrt{n^2 - y^2}\right\}\right\}$$

```
Reduce[x^2 + y^2 == n^2, {x, y, n}]
```

$$n == -\sqrt{x^2 + y^2} \;||\; n == \sqrt{x^2 + y^2}$$

```
Reduce[x^2 + y^2 == n^2, {x, y, n}, Integers]
```

$$(x \mid y \mid n) \in \text{Integers} \;\&\&\; \left(n == -\sqrt{x^2 + y^2} \;||\; n == \sqrt{x^2 + y^2}\right)$$

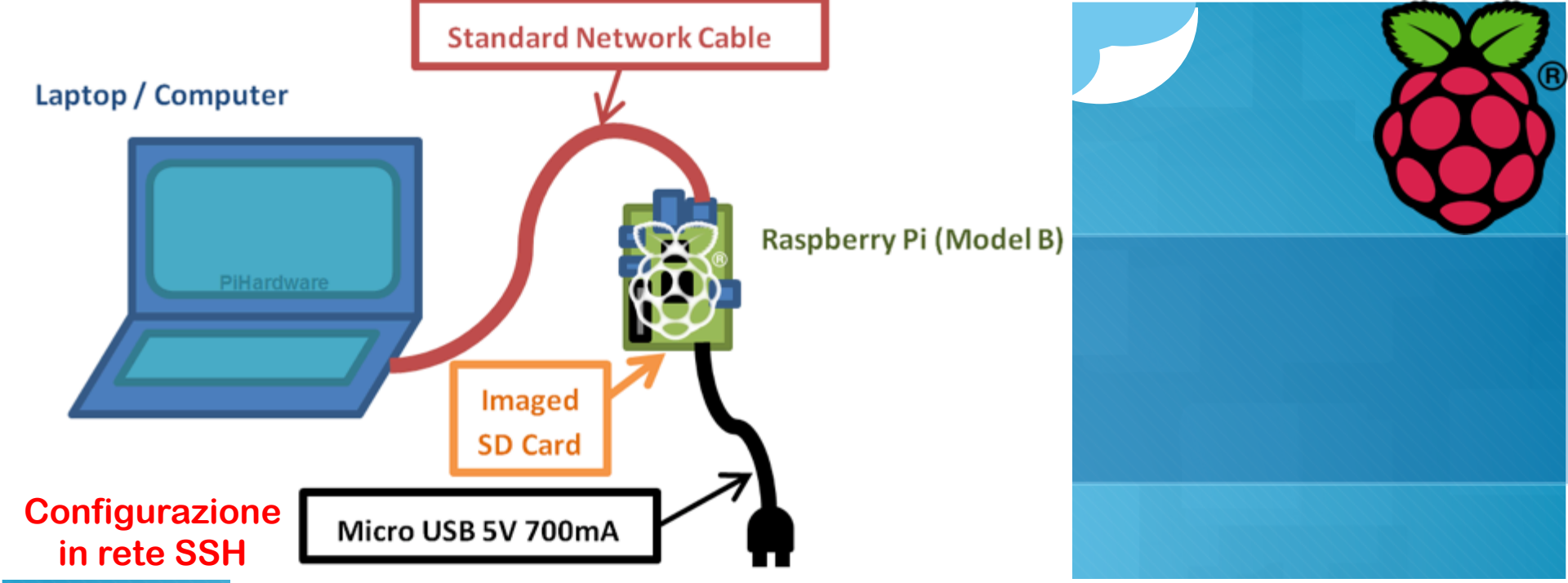
```
Reduce[x^2 + y^2 == 25, {x, y}, Integers]
```

$$(x == -5 \;\&\&\; y == 0) \;||\; (x == -4 \;\&\&\; y == -3) \;||\; (x == -4 \;\&\&\; y == 3) \;||$$
$$(x == -3 \;\&\&\; y == -4) \;||\; (x == -3 \;\&\&\; y == 4) \;||\; (x == 0 \;\&\&\; y == -5) \;||$$
$$(x == 0 \;\&\&\; y == 5) \;||\; (x == 3 \;\&\&\; y == -4) \;||\; (x == 3 \;\&\&\; y == 4) \;||$$
$$(x == 4 \;\&\&\; y == -3) \;||\; (x == 4 \;\&\&\; y == 3) \;||\; (x == 5 \;\&\&\; y == 0)$$

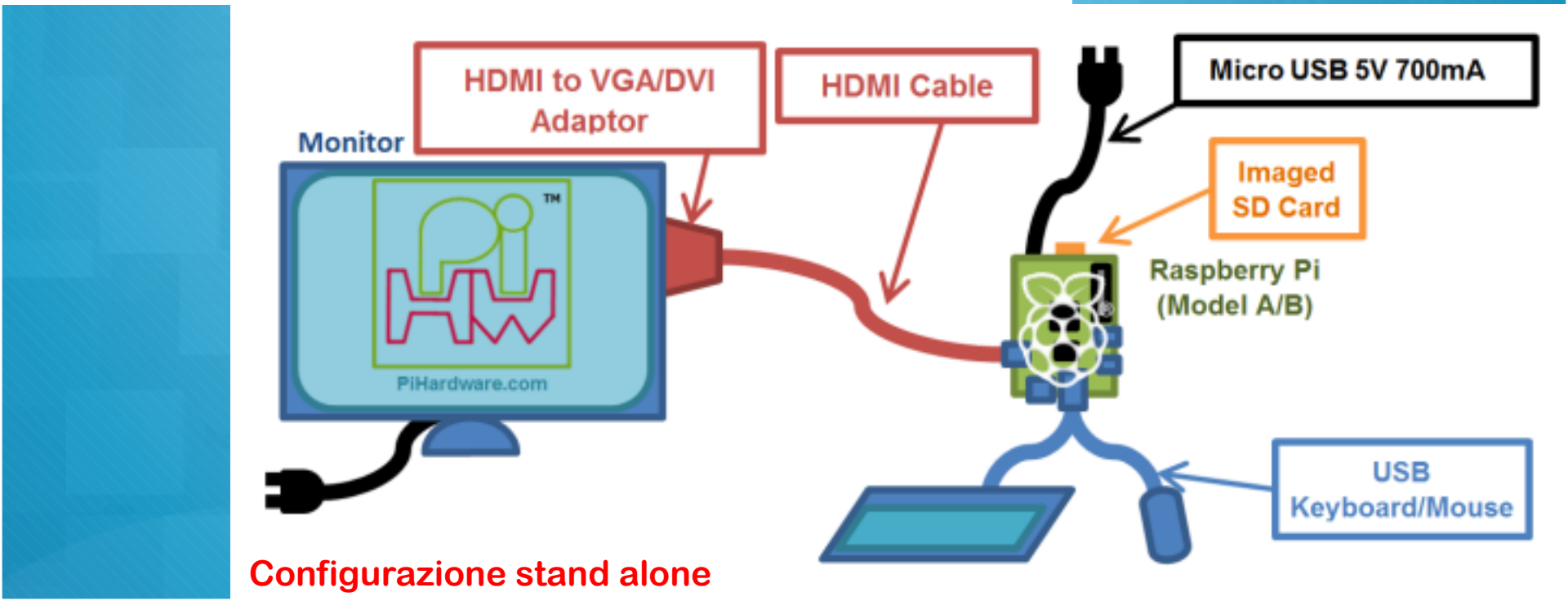
```
Reduce[x^2 + y^2 == n, {x, y, n}, Integers]
```

$$x \in \text{Integers} \;\&\&\; y \in \text{Integers} \;\&\&\; n == x^2 + y^2$$

100%



Configurazione in rete SSH



Configurazione stand alone

In rete con un PC/Mac

- Sul raspberry pi il server SSH è attivo di default
- Il raspberry pi va connesso al PC/Mac via ethernet
- Da terminale sul Mac o con software SSH su PC:
  SSH pi@192.168.2.2
  password = "raspberry"

  ("192.168.2.2", "pi" e "raspberry" sono rispettivamente l'indirizzo ethernet, il nome utente e la password di default del raspberry pi)

Remote Login (ssh)

- **Secure Shell (ssh)** is a network protocol which allows **remote control of Raspberry Pi** from another computer over the network.
- Provides access to the **Command Line** and not to the Full Desktop Environment.
- You can enable or disable the **SSH server** on your Raspberry Pi (it is enabled by default).
- To **Enable SSH**, in the Terminal, enter **sudo raspi-config**, **enable** the **ssh**, finish and exit the menu.
- Use a third-party **SSH client software** on your Windows System (like **puTTy** or **TeraTerm**) and access Raspberry Pi by entering it's I.P. Address, user (pi) and password (raspberry).
- To check Raspberry Pi's I.P. Address, enter **ifconfig** in the terminal.

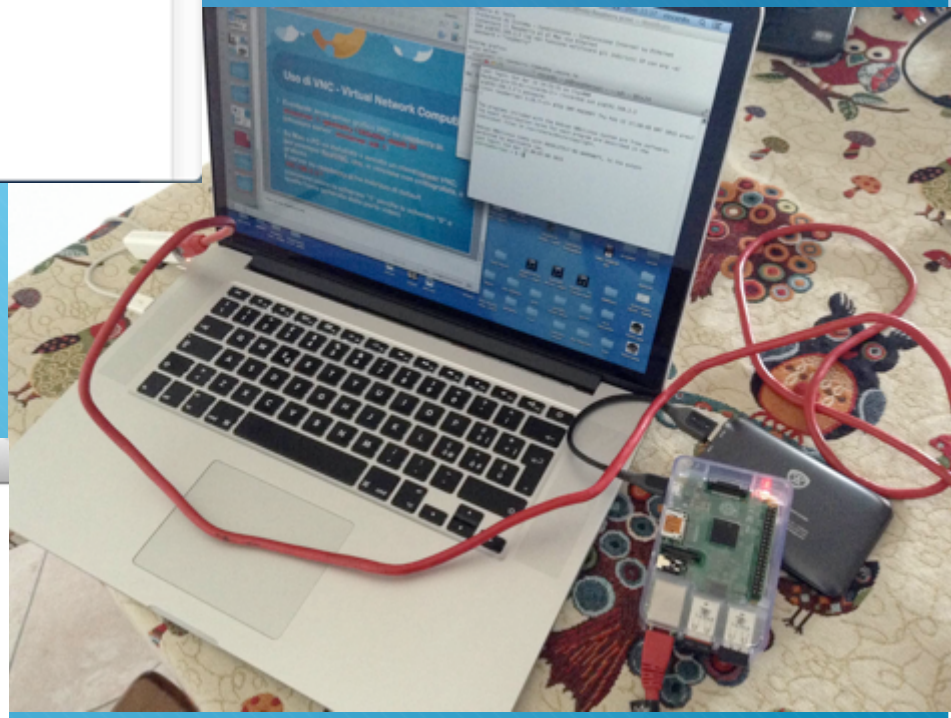
Uso di VNC - Virtual Network Computing

- Eventuale avvio server grafico VNC su raspberry pi:
  vncserver :1 -geometry 1280x854 -depth 24
  (chiusura server: **vncserver -kill :1**)

- Su Mac o PC va installato e avviato un client/viewer VNC, per esempio RealVNC, che, in versione non crittografata, è gratuito.
  Il server su raspberry pi ha indirizzo di default
  192.168.2.2:1
  (conviene usare lo schermo “1” perché lo schermo “0” è quello fisico generato dalle porte video)

riccardo — pi@raspberrypi: ~ — ssh — 80×24

```
Last login: Sun Apr 12 14:25:31 on ttys000
MacBook-pro-15-di-riccardo-2:~ riccardo$ ssh pi@192.168.2.2
pi@192.168.2.2's password:
Linux raspberrypi 3.18.7-v7+ #755 SMP PREEMPT Thu Feb 12 17:20:48 GMT 2015 armv7
l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Apr 12 00:07:06 2015
pi@raspberrypi ~ $ vncserver :1 -geometry 1280x800 -depth 24

New 'X' desktop is raspberrypi:1

Starting applications specified in /home/pi/.vnc/xstartup
Log file is /home/pi/.vnc/raspberrypi:1.log

pi@raspberrypi ~ $
```



VNC Viewer

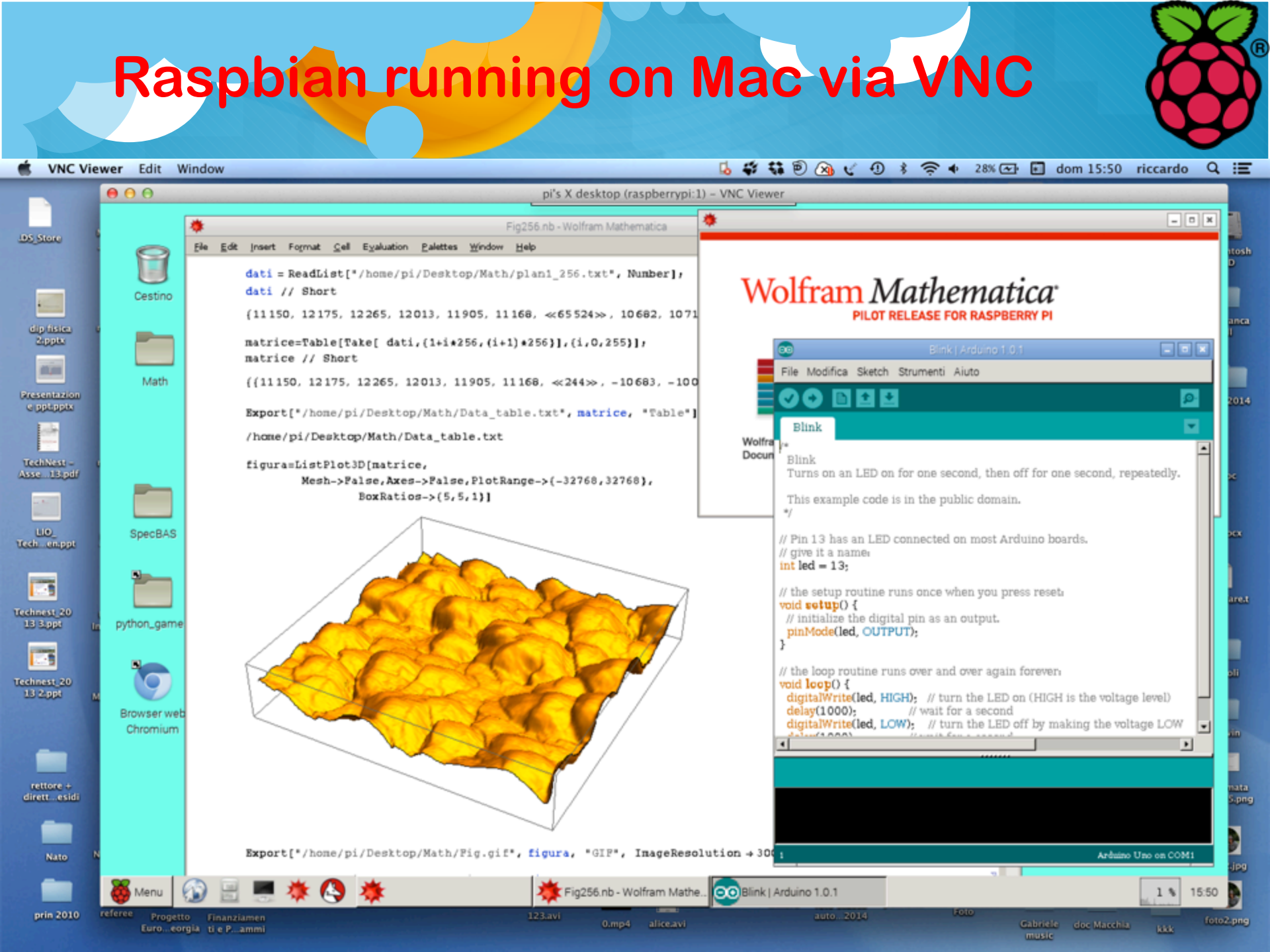
VNC® Viewer

VNC Server: 192.168.2.2:1

Encryption: Let VNC Server choose

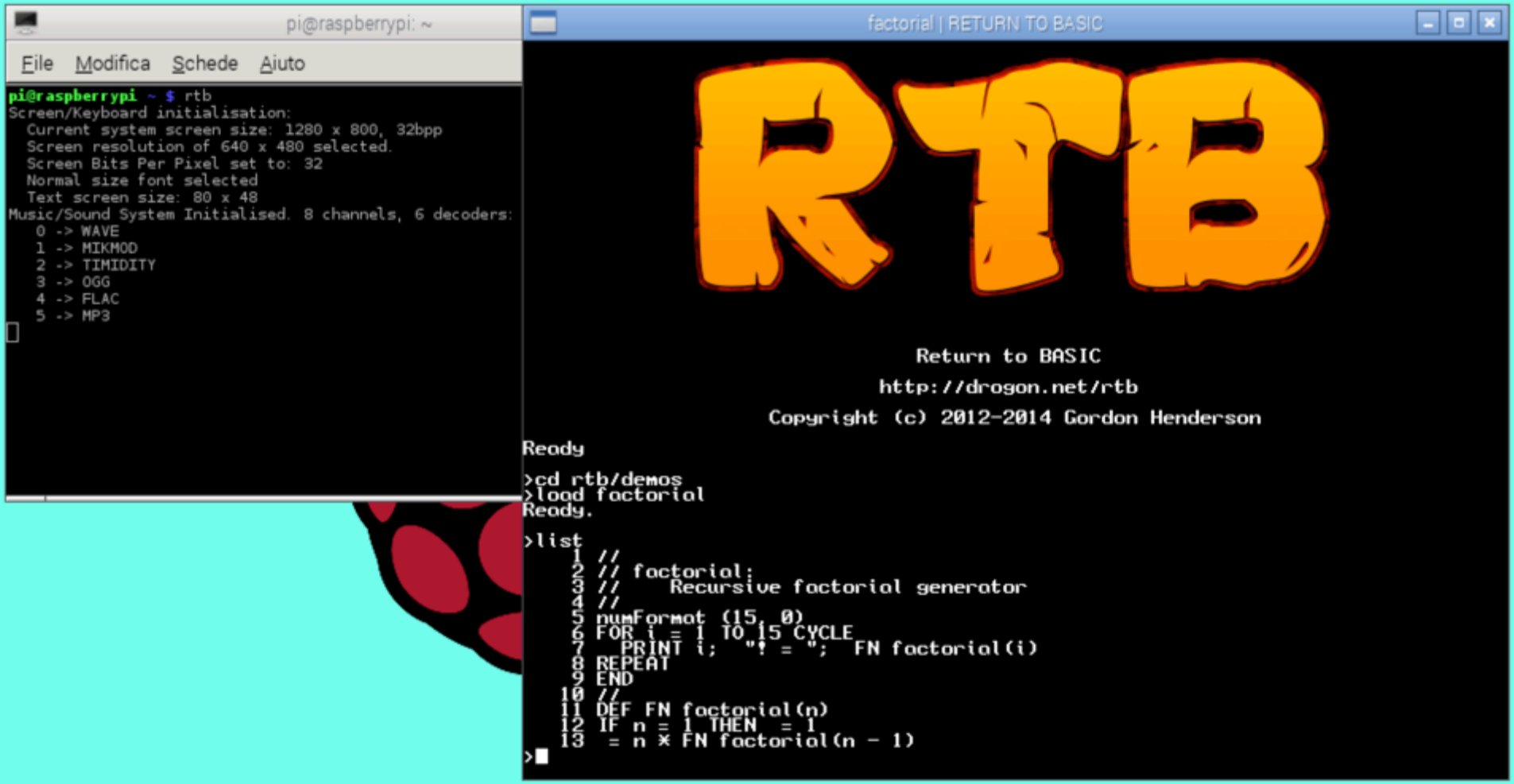
About... Options... Connect

Raspbian running on Mac via VNC



Raspbian programming languages:

- Scratch
- Python
- HTML5
- JavaScript
- JQuery
- C
- C++
- Perl
- Erlang
- Java
- Basic



8 Bit Home Computer Benchmarks

Over the years I've collected quite a few 8 bit home computers. Out of curiosity I decided to write a simple prime sieve benchmark to compare their implementations of BASIC.

```
10 LET W=500:DIM F(W):LET P=1:LET A=3
20 LET F(P)=A:LET P=P+1:IF P>W THEN STOP
30 LET A=A+2:LET X=1
40 LET S=A/F(X):IF S=INT(S) THEN 30
50 LET X=X+1:IF X<P AND F(X)*F(X)<=A THEN 40
60 GOTO 20
```

```
>list
    5 T1 = TIME
   10 W = 500
   11 DIM F(W)
   12 P = 1
   13 A = 3
   20 F(P) = A
   21 P = P + 1
   22 IF P > W THEN GOTO 70
   30 A = A + 2
   31 X = 1
   40 S = A / F(X)
   41 IF S = INT (S) THEN GOTO 30
   50 X = X + 1
   51 IF X < P AND F(X) * F(X) <= A THEN GOTO 40
   60 GOTO 20
   70 PRINT (TIME - T1) / 1000
   80 END
>run
0.098
```

Here are the results from a few of the machines I have to hand:

System	CPU	Time
Acorn Electron	2.0MHz 6502	138
Amstrad CPC464	4.0MHz Z80A	140
Commodore C64	1.0MHz 6510	254
Commodore Plus/4	1.0 MHz 8501	267
Tandy 64K CoCo 2	0.895MHz 6809E	271
Atari 800XL	1.8MHz 6502	316
Sinclair Spectrum +3	3.55MHz Z80A	388

SIEVE Benchmark

Raspberry PI 0,10 sec

in media 2000 volte più veloce dei primi personal computer

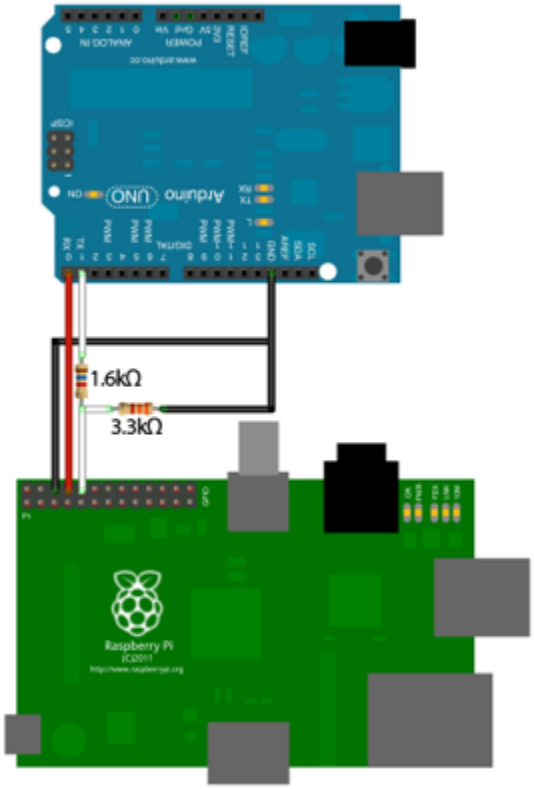
GPIO Programming con RTB

- **PinMode (X,m)**
  X è il PIN indirizzato,
  m è il modo di indirizzamento:
  0 - input; 1 - output; 2 – PWM Output
- **DigitalRead(X)** -> restituisce 0 (low/false) o 1 (high/true)
- **DigitalWrite(X,s)** -> **s** può valere 0 (low) o 1 (high)
- **PwmWrite (X, p)** -> 0 < **p** < 1024

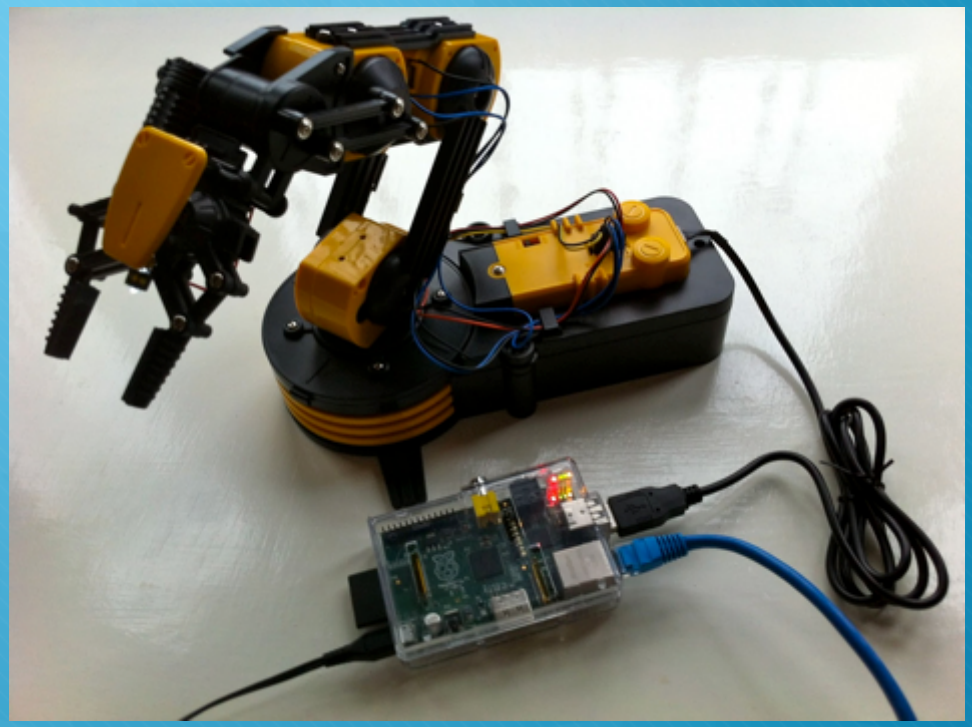
Drogon remote Control Programming con RTB

- Protocollo DRC per il controllo di dispositivi quali i microcontroller, come per esempio Arduino
- **arduino = DRCOpen ("/dev/ttyUSB0")**
- **DrcClose(arduino)**
- **DrcPinMode(arduino, 12, 0)**
  il Pin 12 di Arduino è configurato in Input
  (0 - input; 1 - output; 2 – PWM Output)
- **DrcDigitalRead(Arduino, 12)**
  Legge il Pin 12 di Arduino
- **DrcDigitalWrite(Arduino , 11, 0)**
  Scrive un livello "low" sul Pin 11 di Arduino
- **Voltage = (DrcAnalogRead(Arduino, 4)/1023)*5**
  Legge il voltaggio sul Pin 4
- **DrcPwmWrite(arduino, 5, 200)**

Connessione seriale tra Raspberry pi e Arduino



Robotic Arm Kit con RTB o Fuze Basic



Al Lavoro ...