

# Una tagliatrice laser cinese: facciamola funzionare open!

*Giovanni Mascellani*  
[gio@debian.org](mailto:gio@debian.org)

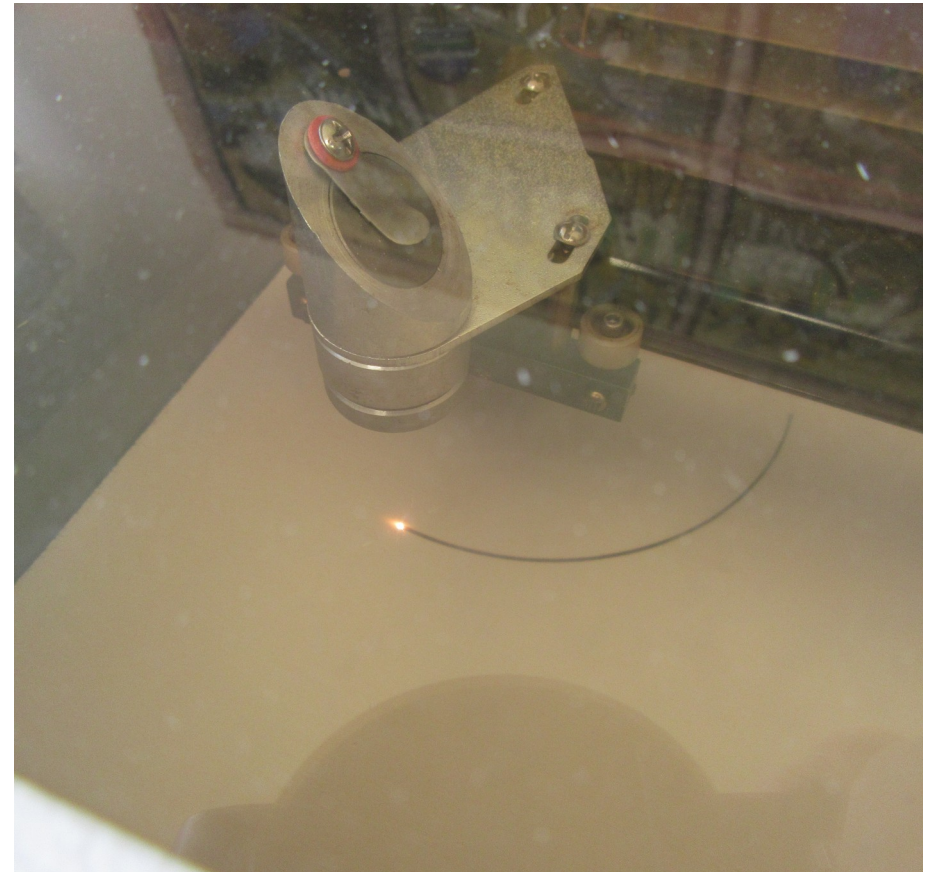
24 ottobre 2015  
Linux Day Pisa

*GULP - Gruppo Utenti Linux Pisa*



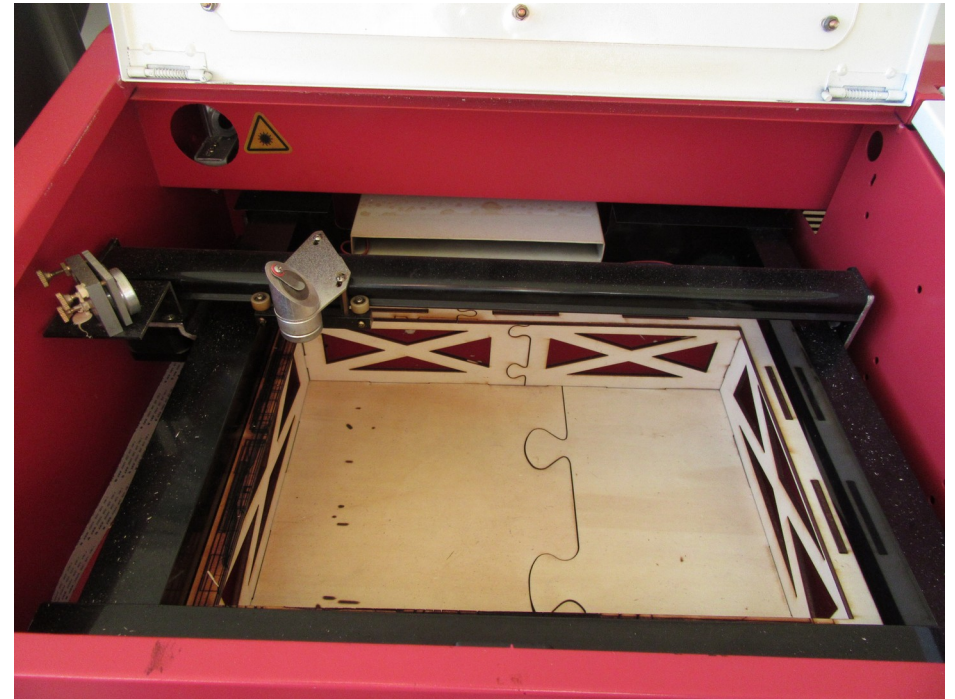
# Cos'è una tagliatrice laser?

- Macchina CNC (Computer Numerical Control) a due assi.
- Laser a CO<sub>2</sub> con luce da circa 10  $\mu\text{m}$ .
- La testina contiene un sistema ottico (lenti e specchi) che direziona e concentra il fascio laser sull'oggetto che sta venendo lavorato. Il materiale viene vaporizzato o bruciato in un punto molto concentrato.

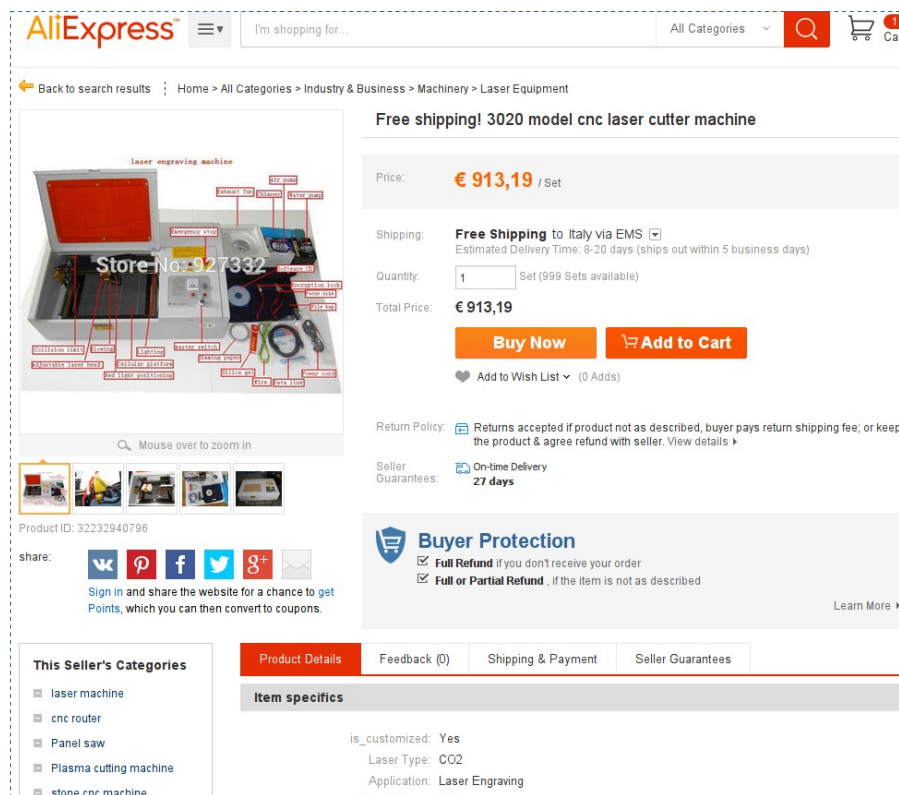


# Cosa ci si può fare

- Può tagliare carta, cartone, compensato, MDF, plexiglass, ...
- Meno possibilità rispetto alla fresatura o alla stampa 3D (si può lavorare solo in 2D).
- Però è anche molto più veloce.
- Gli oggetti 3D si possono costruire assemblando pezzi 2D.



# Sui mercati cinesi...



- Si possono trovare tagliatrici laser di prezzo contenuto (< 1 k€).
- Anche la qualità è molto contenuta...
- Per maggiore sicurezza (costi di importazione, trasporto, ...) è possibile acquistare prodotti già importati in Europa.



# Caratteristiche tecniche

- Area di lavoro: circa 34 x 24 cm.
- Laser da 40 W (forse).
- Meccanica CNC con due stepper NEMA 17.
- Velocità di lavoro regolabile tramite software.
- Potenza di lavoro regolabile tramite hardware.
- Asse verticale non presente (dunque non si può regolare facilmente il punto di fuoco del laser).
- Non sono ancora riuscito a tagliare bene l'MDF. Non ho ancora provato il plexiglass.



# Pros and cons

- Prezzo (relativamente) molto contenuto.
- Struttura molto semplice (quindi è facile metterci le mani sopra).
- Caratteristiche comunque sufficienti a fare lavoretti interessanti.
- Qualità costruttiva bassa (ma non infima).
- Nessun supporto.
- Nessuna certificazione di sicurezza.
- Sistemi di evacuazione fumi e di raffreddamento di bassa qualità.
- Laser da 40 W: sarà vero?
- *Dotazioni di sicurezza basse.*
- **Software di accompagnamento di bassa qualità, piratato, solo per Windows e soprattutto chiuso.**

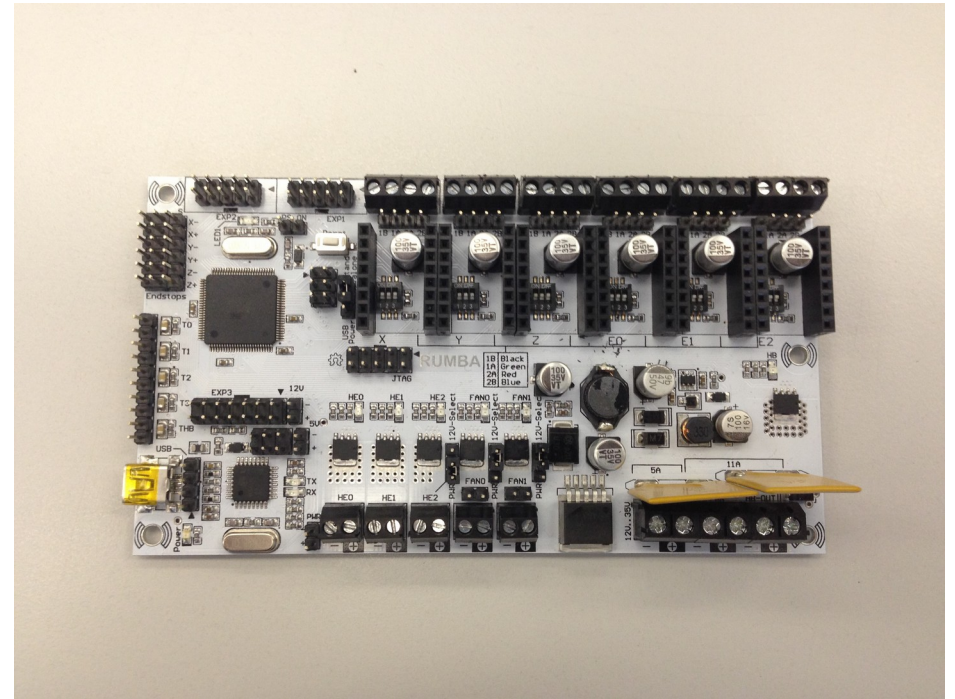
# Rendiamola open e libera!

- Prima possibilità: fare il *reverse engineering* del protocollo USB e scrivere un software compatibile.
  - Lungo e complicato.
  - Si rimane vincolati alla scheda di controllo chiusa.
  - Impossibilità di estendere le funzionalità rispetto a quanto permesso dalla scheda.
- Seconda possibilità: sostituire la scheda con un'altra, riadattando le soluzioni libere che già esistono per le stampanti 3D (RepRap).
  - Riutilizzo del lavoro della comunità.
  - Successive possibilità di estensione e miglioramento.
  - Tutto sommato abbastanza rapido da fare.
  - Non sono il primo, ma non sono riuscito a riutilizzare nessuna delle soluzioni già esistenti.

<http://3dprintzothar.blogspot.it/2014/08/40-watt-chinese-co2-laser-upgrade-with.html>

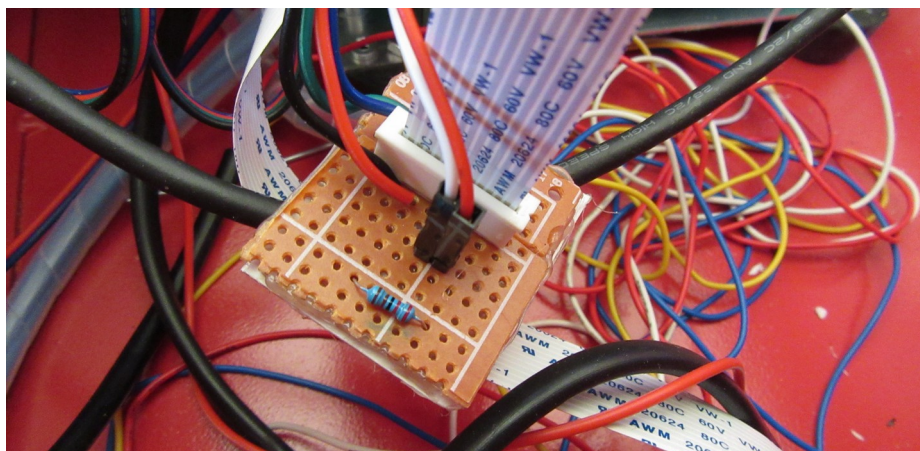
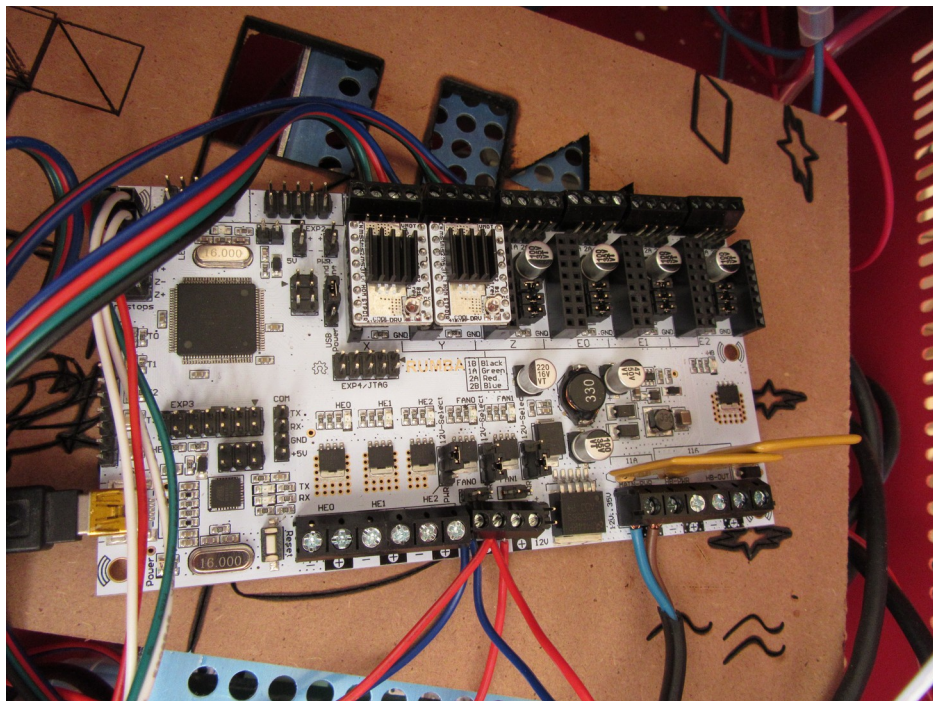
# La scheda di controllo

- Scheda RUMBA.
- È un clone di Arduino progettato apposta per le applicazioni CNC: supporta *on board* sei stepper, sei endstop, due ventole, gestione temperatura.
- Buona parte di queste feature in realtà è inutile per noi.
- Soprattutto funziona a 24 V senza modifiche.





# Wiring della nuova scheda



- Dal punto di vista elettrico: facile, gli stepper, gli endstop, l'alimentazione ed il controllo del laser si interfacciano direttamente sulla scheda RUMBA.
- Unica eccezione: ho dovuto aggiungere un pull-up all'endstop X, perché quello interno di RUMBA era troppo duro.
- Dal punto di vista meccanico: più complicato, ho dovuto cercare un po' di connettori su vari siti, ma alla fine era tutta roba standard.

# Controllo del laser

- Ho attaccato il laser ad uno dei pin che alimentano la ventola.
- Per accendere e spegnere il laser bisogna controllare la ventola (codice M42 P8 su Marlin, preceduto da uno dwell G4).
- Possibilità di modulare la potenza in PWM (però non funziona molto bene).



# Firmware sulla scheda

- Uso una versione modificata del firmware Marlin (che già supporta RUMBA).
- Disabilito il controllo di temperatura, gli assi Z e E e configuro parametri propri della mia macchina (dimensione dell'area di lavoro, struttura degli endstop, parametri degli stepper).
- <https://github.com/geomasc/e/Marlin/commits/Development>

```
340 @@ -340,13 +340,13 @@ const bool Z_MAX_ENDSTOP_INVERTING = false; // se
341 // Disables axis when it's not being used.
342 #define DISABLE_X false
343 #define DISABLE_Y false
344 #define DISABLE_Z false
345 #define DISABLE_E false // For all extruders
346 #define DISABLE_INACTIVE_EXTRUDER true //disable only inactive extruders
347
348 // Invert the stepper direction. Change (or reverse the motor connect
349 #define INVERT_X_DIR false
350 #define INVERT_Y_DIR false
351 #define INVERT_Z_DIR true
352 #define INVERT_E0_DIR true
353 #define INVERT_E1_DIR true
354
355 @@ -366,8 +366,8 @@ const bool Z_MAX_ENDSTOP_INVERTING = false; // se
356 #define X_MIN_POS 0
357 #define Y_MIN_POS 0
358 #define Z_MIN_POS 0
359 #define X_MAX_POS 200
360 #define Y_MAX_POS 200
361 #define Z_MAX_POS 200
362 #define X_MAX_POS 350
363 #define Y_MAX_POS 240
364 #define Z_MAX_POS 200
365
366 //=====
367 @@ -514,7 +514,9 @@ const bool Z_MAX_ENDSTOP_INVERTING = false; // se
368 // default settings
369
370 #define DEFAULT_AXIS_STEPS_PER_UNIT {80,80,4000,500} // default st
371 // Gio Rumba: should be 320 and 320 steps per unit, but I'm correcti
372 // them basing on experimental values
373 #define DEFAULT_AXIS_STEPS_PER_UNIT {320.0 * (120.0 / 121.35), 320.
374 #define DEFAULT_MAX_FEEDRATE {300, 300, 5, 25} // (mm/sec)
375 #define DEFAULT_MAX_ACCELERATION {3000,3000,100,10000} // X, Y, Z, E
```



# Generazione del G-code

```
try:
    t1 /= abs(t1)
except ZeroDivisionError:
    t1_zero = True
try:
    t2 /= abs(t2)
except ZeroDivisionError:
    t2_zero = True
v = p2 - p1
t = t1 + t2

# Handle degenerate cases: when both derivatives are zero, do
# linear interpolation; when just one is zero, ignore it and do a
# simple arc interpolation using the other one
if t1_zero and t2_zero:
    return None, None, None
if t1_zero or t2_zero:
    if t1_zero:
        n = t2 * 1j
    else:
        n = t1 * 1j
    #inkex.errormsg("v = %s, n = %s, scalar = %s" % (v, n, scalar(v, n)))
    try:
        k = 0.5 * abs(p1 - p2) ** 2 / scalar(v, n)
    except ZeroDivisionError:
        return None, None, None
    if t1_zero:
        c2 = p2 + k * n
        if abs(c2) >= 10000.0:
            raise Exception("Bad things happening")
        return None, p1, c2
    else:
        c1 = p1 + k * n
        if abs(c1) >= 10000.0:
            raise Exception("Bad things happening")
        return c1, p2, None

# Compute d, set as d = d2 = d1 (as in "Choosing d1")
sc_vt = scalar(v, t)
sc_tt = scalar(t1, t2)
discr = sc_vt ** 2 + 2 * (1 - sc_tt) * scalar(v, v)
assert discr >= 0.0
if sc_tt != 1.0:
    # Case 1
    d = (discr ** 0.5 - sc_vt) / (2 * (1 - sc_tt))
else:
    sc_vt2 = scalar(v, t2)
    sc_vv = scalar(v, v)
    if sc_vt2 != 0.0:
        # Case 2
        d = sc_vv / (4 * sc_vt2)
    else:
        # Case 3
        pm = p1 + 0.5 * v
```

- Plugin di esportazione per Inkscape.
- Ne esistono un sacco di diverse versioni e fork, tutte con vari bug e feature mancanti.
- Io mi sono scritto la mia versione, con nuovi bug e nuove feature mancanti. ;-)
- Però almeno ci sono le feature che mi servono! (tagliare le zone più interne prima di quelle più esterne; purtroppo anche qui ci sono ancora dei bug)
- Inoltre supporta la mia macchina!
- <https://github.com/giomasce/thlas-inkscape-plugin>

# Evacuazione dei fumi



- Il sistema originale è molto scarso e funziona bene solo nella zona vicina all'evacuatore.
- Ho potenziato l'evacuatore con alcune ventole aggiuntive, che in realtà per ora non fanno tantissimo (in compenso fanno molto rumore).
- C'è una ventola aggiuntiva che “rimescola” l'aria nella camera di lavoro, al solo scopo di impedire al fumo di andare direttamente sulla lente.
- Per ora non c'è nessuna pompa di aria pulita verso l'interno.



# Cosa manca?

- Disabilitare il laser quando il coperchio è aperto.
- Risolvere i bug del plugin per Inkscape e renderlo più utilizzabile e configurabile.
- Implementare l'incisione raster (sia nel plugin per Inkscape che nel firmware; esistono già alcune implementazioni non mantenute).
- Ottimizzare l'ottica (specchi e lenti)?
- Misurare la potenza effettiva del laser.
- Migliorare ulteriormente l'evacuazione dei fumi (troppo fumo nell'ambiente di lavoro sporca lenti e specchi).
- Documentare e pubblicare il tutto.
- Sicurezza/utilizzabilità in ambienti di lavoro?
- Calibrazione fine del movimento e controllo di qualità e precisione.

# Grazie! Domande?

Ringrazio Salvatore Balestrino,  
Massimo Pisani e [opensourcehardware.it](http://opensourcehardware.it)  
per i suggerimenti e l'aiuto.

*Le informazioni ed il codice presente sono solo  
esemplificativi. Se li usate lo fate a vostro rischio  
e pericolo. Non giocate mai con un laser da 40 W  
senza sapere cosa fate, è molto pericoloso!*