

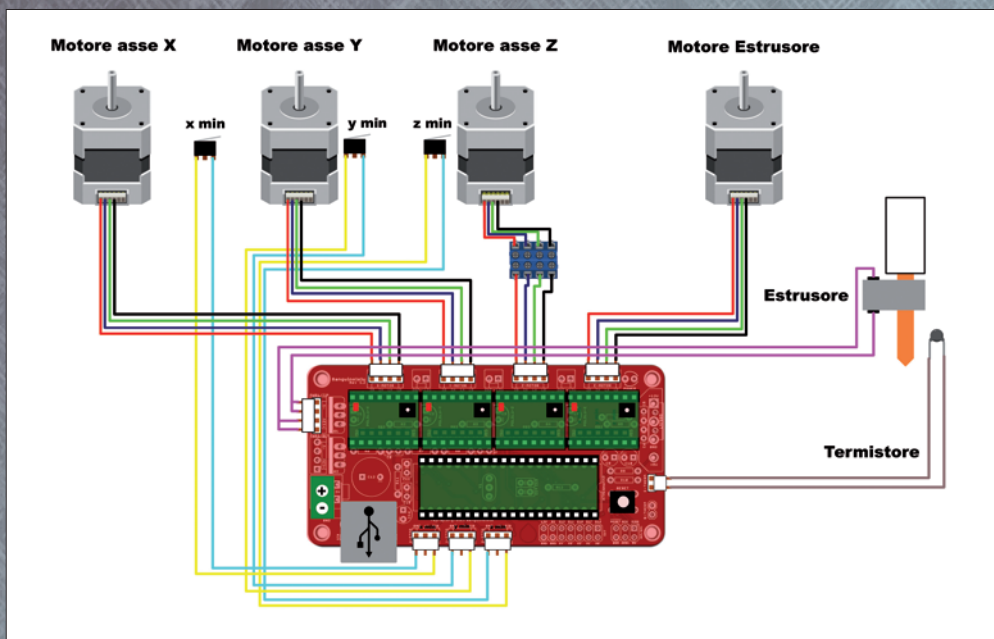
# 3DRAG SENZA SEGRETI

Impariamo a conoscere le caratteristiche hardware e software della stampante 3D targata Elettronica In.

di GABRIELE DAGHETTA  
testo di SIMONE MAJOCCHI

**F**ra i vantaggi dell'hardware e del software Open Source ci sono la possibilità di conoscere tutti i dettagli e l'accesso a tutti i parametri per modificare il funzionamento del dispositivo. Quante volte, guardando un oggetto, ci è venuto in mente che si poteva migliorare il suo funzionamento facendo questa o quella modifica? Con i prodotti commerciali la modifica invalida la garanzia e richiede del reverse engineering prima di capire come intervenire e questa attività potreb-

be addirittura violare la Legge sulla tutela della Proprietà Intellettuale (la cosiddetta IP in inglese). Con i prodotti Open Source, invece, abbiamo già a disposizione tutte le informazioni e il sorgente grazie ai quali studiare e implementare le nostre modifiche nel modo migliore. Quello che ci viene chiesto in cambio è l'impegno morale a ricondividere il nostro lavoro con la comunità, così da mantenere il circolo virtuoso dello sviluppo e del miglioramento gratuiti e condivisi.



*Lo schema dei collegamenti elettrici dalla scheda Sanguinololu ai motori, ai microswitch e ai componenti attivi della punta di estrusione.*

3Drag appartiene alla famiglia dell'Open Source ed è quindi realizzata per poter beneficiare da un lato dei nuovi sviluppi software e dall'altro per poter essere d'ispirazione ad altri nella realizzazione di stampanti RepRap. Rispetto al concetto originale, 3Drag non produce i pezzi per replicare se stessa, ma questo è una conseguenza dell'approccio "industriale" alla realizzazione delle parti, mentre tutte le informazioni relative alle parti meccaniche sono disponibili per consentire a chi vuole di usare una CNC e il proprio materiale di realizzarsi autonomamente una 3Drag.

### IL FIRMWARE

Il software residente sulla scheda Sanguinololu di 3Drag è responsabile della gestione delle funzioni di base della stampante e interpreta i vari comandi del G-Code per trasformarli in movimenti del motore e dell'estrusore.

Questo codice, disponibile come sorgente per tutte le versioni rilasciate in Open Source ed elencate anche sul nostro sito "<http://3dprint.elettronica.in.it>" è realizzato dai vari autori rispettando alcune convenzioni del progetto RepRap, ovvero la possibilità di definire attraverso un file di configurazione le caratteristiche meccaniche ed elettroniche della stampante.

Ogni firmware prevede quindi la struttura basata su tre motori – uno per asse di spostamento – un motore per estrusore, il riscaldatore assieme al suo termistore di controllo, più un piatto riscaldato, le ventole di raffreddamento e i microswitch per la regolazione della posizione di inizio o fine corsa. A questo poi gli autori aggiungono funzionalità specifiche come il supporto del lettore di schede SD,

i pulsanti o lo schermo di controllo LCD, oppure dedicano lo spazio disponibile per il codice ad algoritmi per l'ottimizzazione degli spostamenti o alla ricostruzione degli archi per migliorare la stampa. Il microcontrollore, è fondamentale ricordare, ha uno spazio di memoria limitato e quindi non è possibile aggiungere funzioni e algoritmi al firmware senza tenere conto di questo limite. Una soluzione a questo è l'uso di microcontrollori con più spazio di memoria, come ad esempio l'ATmega 1284P al posto dell'ATmega 644P: fra i due la compatibilità è pin to pin, mentre le dimensioni della memoria passano da 64 a 128K di Flash, da 4K a 16 K di RAM e da 2K a 4K di EEPROM. Questo microcontrollore può anche passare dai 16 MHz del 664P ai 20 MHz per una esecuzione più veloce del codice.

Tornando alla scheda e al controller che interfaccia motori e microswitch, le modalità di funzionamento effettivo di ciascuna di queste parti dipende dalle loro caratteristiche elettriche e meccaniche e per questo il file di configurazione permette di specificare i dati che altrimenti potrebbero essere ambigui.

### LA CONFIGURAZIONE DI 3DRAG

In questa sezione vi spiegheremo tutte le voci specifiche per l'hardware e il funzionamento di 3Drag facendo riferimento alle variabili utilizzate nei file di configurazione dei firmware. Dato che abbiamo scelto Marlin come nostro Firmware di riferimento, le variabili saranno quelle di questo firmware, ma il loro nome e la descrizione vi aiuteranno a trovare le variabili omologhe negli altri firmware, nel caso in cui decidiate di sperimentarli.

Lo stesso discorso vale per aggiornare Marlin a una release successiva a quella fornita. Il processo di

aggiornamento è spiegato in una sezione specifica, successiva a questa.

Iniziamo con il dire che ogni firmware deve conoscere il tipo di scheda utilizzata: nel nostro caso è una Sanguinololu 1.2, individuata solitamente con il codice 62 negli elenchi di schede:

```
/ Sanguinololu 1.2 and above = 62
#define MOTHERBOARD 62
```

Con questa informazione, il firmware viene compilato con la corretta configurazione di pin, assegnando i pin digitali e quelli analogici alle varie risorse hardware come i moduli di potenza degli stepper e gli switch di inizio/ fine corsa.

L'errata configurazione di questo valore determina un funzionamento errato della stampante. Riportiamo lo schema del cablaggio di questa scheda così che possiate avere un riferimento a livello di collegamenti elettrici sulla scheda, motori e microswitch. Il parametro successivo riguarda il tipo di NTC utilizzata: i termistori si caratterizzano per alcuni parametri specifici (i coefficienti A, B e C), da utilizzare nell'equazione di Steinhart-Hart che permette una certa accuratezza nel passaggio da valore espresso in ohm a temperatura misurata. Se i coefficienti non sono corretti, la temperatura si può discostare in modo apprezzabile da quella reale, e per la fusione del filamento nel modo corretto questo va evitato. Con 3Drag utilizziamo una NTC da 100K e non abbiamo il piano di stampa riscaldato a cui va abbinato un altro sensore di tipo NTC. Per questi motivi nell'area destinata alle impostazioni dei sensori dobbiamo avere il tipo 5 per il Temp\_Sensor\_0 e dobbiamo anche verificare che non ci sia alcun sensore associato al piano (Bed\_Sensor\_0) altrimenti il firmware attenderebbe che anche questo abbia raggiunto la temperatura e la stampa non avrebbe mai inizio. I dati relativi ai sensori sono solitamente contenuti in una sezione specifica, come quella riportata qui di seguito.

```
//===== Thermal Settings=====
//===== Thermal Settings=====
//===== Thermal Settings=====
// 5 is 100K thermistor - ATC Semitec
// 104GT-2 (4.7k pullup)
#define TEMP_SENSOR_0 5
...
#define BED_SENSOR_0 0
```

La sezione successiva su cui dobbiamo agire è quella relativa alla logica dei microswitch di inizio corsa, ovvero quei tre interruttori che permettono al firmware di individuare l'inizio dell'area utile per i due assi X e Y, assieme al punto da cui iniziare a depositare sul piatto di stampa il materiale (asse Z).

La flessibilità del firmware al proposito è massima: i microswitch potrebbero essere anche posizionati a fine corsa e non all'inizio, facendo fare al firmware i conti per il posizionamento a 0,0,0. Allo stesso modo l'interruttore può essere normalmente aperto o chiuso.

Noi abbiamo scelto di usare la soluzione "Normalmente Chiuso" così da bloccare immediatamente il motore nel caso il collegamento sia interrotto o l'interruttore sia difettoso. Questa modalità richiede il valore booleano "false" alle quattro costanti che comunicano al firmware se la logica è invertita (la nostra infatti non lo è, e l'interruttore apre il circuito quando viene azionato). In generale, se un interruttore ha un ruolo importante, come il fermo di un motore a inizio o fine corsa, oppure un blocco di sicurezza quando azionato, è buona norma utilizzarlo nella modalità NC (normalmente chiuso) così da avere una protezione rispetto al suo malfunzionamento e all'interruzione dei collegamenti. Diversamente un difetto o un interruttore scollegato non potrebbe arrestare il motore a fine corsa o l'intero dispositivo in caso di emergenza.

Un altro elemento scelto come opzione è l'uso di interruttori meccanici: alcuni modelli utilizzano dei sensori ottici alimentati, mentre quelli meccanici per la loro natura richiedono solo una resistenza di pull up per eliminare eventuali disturbi sugli ingressi ad alta impedenza. Le costanti che attivano le resistenze sono solitamente già impostate con il valore corretto.

```
const bool X_ENDSTOPS_INVERTING = false; // set to true to invert
// the logic of the endstops.
const bool Y_ENDSTOPS_INVERTING = false;
const bool Z_ENDSTOPS_INVERTING = false;
```

Definita la logica, va ora comunicata al firmware la posizione dell'interruttore: relativamente alla ricerca della posizione di 0 di ciascun asse questo può essere a inizio o a fine corsa. Se è all'inizio (valore "-1" ovvero MIN) il firmware sposterà il piatto di stampa verso la posizione dell'inizio asse (0) e si arresterà quando viene azionato l'interruttore. Se l'interruttore fosse impostato e posizionato per il valore massimo dell'asse (alla fine) tramite il valore "1", il firmware sposterebbe il piatto verso il fine corsa per attivare l'interruttore e poi tornerebbe indietro della lunghezza impostata per quello specifico asse. Per questioni di praticità nella regolazione delle coordinate X=0, Y=0 e della taratura delle viti di regolazione fine, gli interruttori sono posizionati all'inizio della corsa degli assi e quindi tutti i valori delle tre costanti sono impostati a "-1"

```
// ENDSTOP SETTINGS:
// Sets direction of endstops when homing;
// 1=MAX, -1=MIN
#define X_HOME_DIR -1
#define Y_HOME_DIR -1
#define Z_HOME_DIR -1
```

Come indicavamo in precedenza, nella ricerca della posizione "home" che corrisponde allo "0" per ciascun asse, il firmware tiene comunque in considerazione le misure espresse in millimetri per ciascuno degli assi, così da evitare i danni dovuti a spostamenti oltre il limite fisico della meccanica. Dato che utilizziamo solo gli interruttori di "inizio corsa" e non una coppia inizio/fine per ciascun asse, il firmware richiede dei valori di minimo e massimo oltre i quali, una volta fatto l'azzeramento, i motori si fermano. Nel caso di 3Drag, il limite è praticamente uguale al volume di stampa e l'aggio meccanico è in ragione di circa un centimetro per parte. Impostare i valori minimi e massimi a quelli dell'area di stampa evita quindi di rischiare il blocco per raggiunti limiti fisici di spostamento. Con altre architetture meccaniche di stampa, questi valori potrebbero essere anche di qualche centimetro superiori all'area di stampa utilizzabile e questo giustifica la necessità di inserire nel firmware dei valori adatti all'hardware specifico. Da notare che prima di aver "informato" il firmware dell'effettiva posizione del piano di stampa e dell'estrusore, non ci sono blocchi software allo spostamento oltre questi valori minimi e massimi proprio perché non ci sono riferimenti affidabili. Evitate quindi di spostare con i comandi manuali la meccanica prima di aver dato il comando di home per ciascun asse (oppure quello generale di home che azzerava in sequenza X, Y e Z). Per 3Drag, quindi, i valori sono per X e Y da 0 a 200, mentre Z può arrivare a 220 mm.

```
// Travel limits after homing
#define X_MAX_POS 200
#define X_MIN_POS 0
#define Y_MAX_POS 200
#define Y_MIN_POS 0
#define Z_MAX_POS 220
#define Z_MIN_POS 0
```

Eccoci ora alle pulegge e alle cinghie che, a seconda di come sono montate, determinano lo spostamento del piatto di stampa e dell'estrusore in un verso o nell'altro con la rotazione oraria o antioraria di ciascun motore. Basta montare un motore da una parte della struttura piuttosto che dall'altra per invertire il senso di spostamento, come basta fissare la cinghia prendendola sopra o sotto per invertire la direzione. In pratica il piatto si sposta in un senso o nell'altro in base alla combinazione di tutti questi elementi e ovviamente il progettista del firmware

ha fatto in modo che la cosa potesse essere gestita con semplicità, motore per motore. Il parametro è stato denominato "Invert\_<asse>\_Dir" per semplicità e può essere vero o falso. Con 3Drag la combinazione di posizione motore / lato puleggia / aggancio cinghia porta i tre assi X, Y e Z ad essere impostati come "false", ovvero non sono da invertire come movimento, mentre l'estrusore va invertito in quanto c'è un piccolo ingranaggio che aziona la grande ruota dentata a cui è fissato il bullone zigriano e questo meccanismo determina l'inversione del senso di rotazione. Si tratta di una situazione comune alla maggior parte degli estrusori di tipo Wade con il motore sul lato sinistro.

```
#define INVERT_X_DIR false //
#define INVERT_Y_DIR false //
#define INVERT_Z_DIR false //
#define INVERT_E0_DIR true //
#define INVERT_E1_DIR true //
#define INVERT_E2_DIR true //
```

Ultimo parametro legato alla meccanica su cui vale la pena soffermarsi è quello che determina il comportamento dei motori quando non vengono azionati per il movimento, durante tutto il processo di stampa. I motori stepper, se alimentati, restano bloccati sulla loro posizione ed esercitano una significativa resistenza al movimento; questo però ha come effetto collaterale un significativo assorbimento di corrente e il conseguente riscaldamento del motore e dell'elettronica di controllo.

Quando si stampa ha quindi senso mantenere i due assi X e Y attivi, mentre lo Z può essere disabilitato (non alimentato) per tutto il tempo in cui viene disegnato lo strato. Anche l'estrusore va tenuto alimentato per evitare che la pressione del materiale plastico possa far arretrare il filamento durante gli spostamenti senza stampa. Comunque, dopo un certo periodo di inattività a stampa ultimata, i motori vengono disabilitati per non surriscaldarli e questo intervallo fa parte delle impostazioni dell'applicazione di stampa.

```
// Disables axis when it's not being used.
#define DISABLE_X false
#define DISABLE_Y false
#define DISABLE_Z true
#define DISABLE_E false // For all extruders
```

Come ultimo parametro specifico per la meccanica di 3Drag abbiamo il numero di step che ciascun motore deve compiere per effettuare uno spostamento di una unità di misura (nel nostro caso 1 mm). Il calcolo tiene conto del diametro della puleggia, oppure del passo della vite o ancora del rapporto fra ingranaggi e diametro del bullone dentato di

estrusione. Il tutto va anche elaborato in base alle impostazioni della scheda e dei driver che possono essere gestiti a passo pieno oppure a 4, 8 o 16 microstep; sulla 3Drag tutti i motori vengono gestiti a 16 microstep, portando una rotazione completa dell'albero a 3200 microstep. Senza complicarvi troppo la vita, vi forniamo direttamente i conteggi che sono pari a 64.25 per X e Y, 2560 per Z e 654 per l'estrusore. Per quest'ultimo i conteggi sono stati effettuati in base all'effettiva lunghezza del filo utilizzato rispetto a quello previsto dal software di slicing, trovando il valore che più si avvicina alla realtà.

```
// default settings
#define DEFAULT_AXIS_STEPS_PER_UNIT {64.25,64.25,2560,654} //
```

Inizia ora la sezione delle velocità e delle accelerazioni: i valori indicano i massimi che possono essere applicati dal firmware e servono per far funzionare la meccanica nel modo più efficace e veloce possibile, senza però mettere la struttura in risonanza o portarla oltre il limite fisico introducendo errori e distorsioni nella stampa.

Il primo set di valori da definire riguarda le velocità lineari massime per gli assi e l'estrusione. Nel caso di 3Drag il valore può essere portato a 500 mm/sec (ovvero uno spostamento di mezzo metro al secondo, pari a 30 metri al minuto) sui due assi del piano di stampa e sull'estrusore, ma non è detto che questi valori possano essere raggiunti per qualsiasi stampa e condizione. L'asse Z va invece limitato a 50 mm/sec in quanto è molto demoltiplicato (2560 step per fare 1 mm) e una velocità superiore a 5 cm in un secondo richiederebbe la rotazione del motore oltre le sue normali specifiche di funzionamento.

```
#define DEFAULT_MAX_FEEDRATE {500, 500, 50, 500} // (mm/sec)
```

La velocità va poi gestita con un valore di accelerazione espresso in mm/secondo\*secondo che porta il motore da fermo alla velocità richiesta. Evidentemente non è conveniente avere il motore che parte e si ferma in modo istantaneo in quanto comporterebbe sollecitazioni notevoli a tutta la struttura meccanica (pensate a quello che succede quando in auto vi scappa il piede dalla frizione e partite di "botto"), ma allo stesso modo dato che le stampe sono spesso fatte da piccoli spostamenti, avere un valore di accelerazione basso potrebbe non consentire ai motori di arrivare alla velocità prevista se non in rari casi. Accelerazioni troppo elevate, invece, potrebbero portare la struttura a vibrare e a gestire male i movimenti piccoli e alternati (un fill di una parete stretta ad esempio), quindi si tratta di un valore da mettere a punto anche in base alle tipologie di lavori che si intendono stampare.

```
// X, Y, Z, E maximum start speed for accelerated moves.
#define DEFAULT_MAX_ACCELERATION {9000,9000,100,10000}
```

I parametri di accelerazione standard per la stampa e la ritrazione del filo sono invece definiti in due linee specifiche e dalle nostre prove il valore di 1000 si adatta bene alla struttura.

```
// X, Y, Z and E max acceleration in mm/s^2 for printing moves
#define DEFAULT_ACCELERATION 1000
```

```
// X, Y, Z and E max acceleration in mm/s^2 for r retracts
#define DEFAULT_RETRACT_ACCELERATION 1000
```

## LA PROGRAMMAZIONE DEL FIRMWARE

La stampante 3Drag viene fornita con il firmware già programmato sulla scheda, sia che abbiate acquistato il kit, sia che vi sia arrivata montata e collaudata. Il firmware scelto è Marlin e sul sito [3dprint.elettronica.in](http://3dprint.elettronica.in) potete trovare nella sezione software sia la versione corrente in uso presso i nostri laboratori, sia i collegamenti alle versioni più recenti rilasciate per questo firmware.

La programmazione della scheda si effettua tramite il medesimo collegamento USB utilizzato per pilotare la stampante e questa è una piacevole eredità che la compatibilità con Arduino ci offre. Marlin – come gli altri firmware – è infatti nato come codice scritto su IDE di Arduino e, infatti, quando lo si deve aggiornare si deve proprio usare l'IDE in cui carichiamo il codice sorgente con tutti i moduli interamente accessibili e commentati.

Al momento in cui scriviamo, per una serie di moti-

**Cosa usiamo con 3Drag**

Firmware scheda:	Marlin versione RC3 - 72
Software di stampa:	Repetier Host versione 0.74
Software Slicing:	Slic3r versione 0.9.3

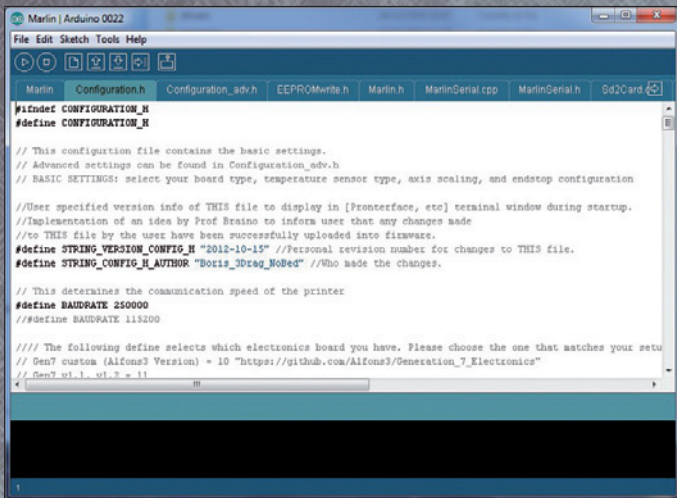
**Aggiornamenti software**

Ecco le versioni più recenti disponibili sui siti degli sviluppatori dei software che abbiamo scelto di utilizzare con la nostra 3Drag.

Marlin:	16 settembre 2012 - V 1 RC
Repetier-Host:	4 settembre 2012 - setupRepetierHost_0_72.exe
Slic3r:	30 agosto 2012 - 0.9.2

- **Firmware per le stampanti 3D**  
La scheda che controlla 3Drag utilizza un programma software - il firmware - per interpretare i comandi di stampa. **Molto importante: la stampante viene fornita già completa di firmware MARLIN preinstallato.**
- **Software di stampa**  
L'invio dei file alla stampante 3D avviene tramite appositi programmi che, oltre alla semplice comunicazione, offrono numerose funzionalità aggiuntive.

*Sul sito [3dprint.elettronica.in](http://3dprint.elettronica.in) si trova la sezione Software con le indicazioni delle release utilizzate per 3Drag.*



L'IDE 022 di Arduino permette di riprogrammare la scheda con il firmware. Questo va configurato per essere compatibile con le caratteristiche di 3Drag intervenendo sul file "Configuration.h".

vi pratici, si deve utilizzare la versione 022 dell'IDE, disponibile all'indirizzo [arduino.cc/en/Main/Software](http://arduino.cc/en/Main/Software) nell'area "Previous IDE Versions".

Il file sorgente con la configurazione per 3Drag la potete scaricare dal nostro sito sopra indicato e una volta aperto potrete vedere tutta una serie di schede sull'IDE.

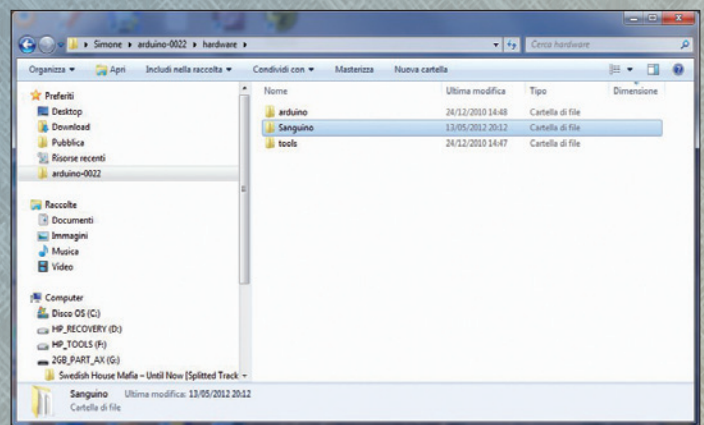
La seconda scheda è quella denominata "Configuration.h" e contiene i vari parametri che abbiamo indicato nelle pagine precedenti. Se effettuate delle modifiche è importante ricordare che dopo aver salvato, avrete una versione in cui non ci sono più i valori da noi consigliati. Conservate quindi il file originale zippato oppure fate una copia di tutta la cartella Marlin e poi datele un nuovo nome e aprite Marlin.pde dalla nuova cartella.

Per programmare la scheda Sanguinololu da IDE

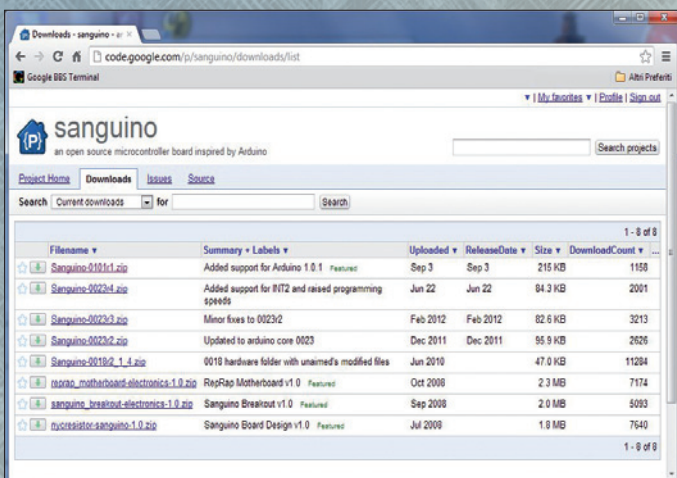
di Arduino è necessario aggiungere la prima volta i file relativi all'hardware. Trovate il necessario sul sito di Sanguino all'indirizzo [code.google.com/p/sanguino/downloads/list](http://code.google.com/p/sanguino/downloads/list) e il contenuto dello ZIP che avrete scaricato va inserito nella cartella "hardware" di Arduino IDE.

Se avete collocato i file nella cartella corretta, come indicato nell'immagine, troverete sotto il menu delle schede di Arduino il nuovo set di schede Sanguinololu, con le varie opzioni di CPU e Clock.

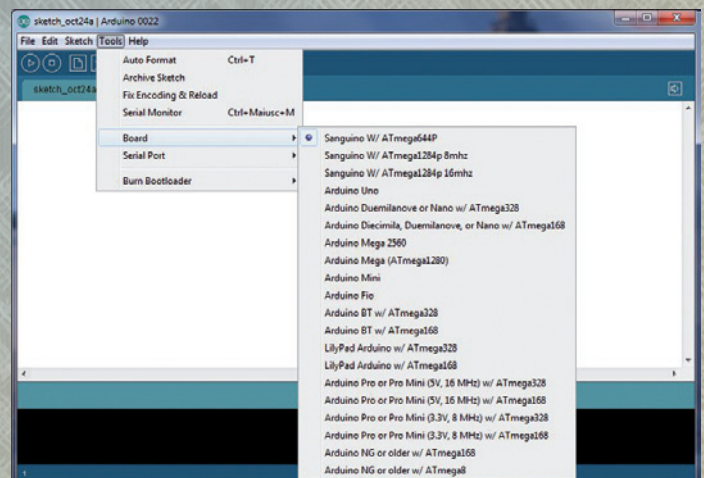
L'attuale versione montata sulla 3Drag è ATmega 644P e dovrebbe trovarsi all'inizio dell'elenco; se non trovate la scheda, avete sbagliato la cartella in cui avete messo i file scaricati dal sito Sanguino. Se scegliete una tipologia non corrispondente a quella montata, in fase di programmazione avrete un messaggio di errore e non avrete fatto alcun danno. Ovviamente, prima di programmare, dovrete an-



Il contenuto dello ZIP scaricato da Sanguino va inserito come riportato nell'immagine.



La scheda Sanguinololu richiede un apposito set di file da posizionare nella cartella Hardware dell'installazione dell'IDE. Vanno scaricati dal sito Sanguino.



I file di Sanguino rendono disponibili delle nuove schede nell'elenco delle Board supportate. Per 3Drag va scelta "Sanguino W/ ATmega644P".

```

Marlin | Arduino 0022
File Edit Sketch Tools Help
Marlin Configuration.h Configuration_adv.h EEPROMwrite.h Marlin.h MarlinSerial.cpp MarlinSerial.h Sd2Card.h

//Reprap M Codes
// M0 - Unconditional stop - Wait for user to press a button on the LCD (only if ULTRA_LCD is enabled)
// M1 - Same as M0
// M104 - Set extruder target temp
// M105 - Read current temp
// M106 - Fan on
// M107 - Fan off
// M109 - Wait for extruder current temp to reach target temp.
// M114 - Display current position

//Custom M Codes
// M17 - Enable/Power all stepper motors
// M18 - Disable all stepper motors; same as M84
// M20 - List SD card
// M21 - Init SD card
// M22 - Release SD card
// M23 - Select SD file (M23 filename.g)
// M24 - Start/Resume SD print
// M25 - Pause SD print

```

L'elenco più aggiornato dei comandi supportati dal firmware è presente nella parte iniziale del file Marlin.pde.

che scegliere la porta COM da utilizzare nel menu "Serial Port", assicurandovi che non ci sia nulla che sta colloquiando con la stampante (ad esempio il software di stampa) e che il ponticello per la programmazione sia inserito sulla scheda Sanguinololu (lo trovate a metà del microcontrollore, dal lato dei driver).

Tutte le schede da noi utilizzate sono state programmate con questo sistema e hanno un bootloader compatibile Arduino opportunamente programmato e "collaudato" con un Arduino IDE 022. Vi segnaliamo infine che nella prima parte del file Marlin.pde si trova l'elenco dei comandi gestiti dal firmware sia come G-Code standard, sia come comandi proprietari con tutta la documentazione necessaria al loro utilizzo.

### DEFINIRE I PARAMETRI VIA SOFTWARE

Uno dei motivi per i quali abbiamo scelto di montare il firmware Marlin è la sua possibilità di accedere attraverso una serie di comandi specifici ai vari parametri di funzionamento. In pratica Marlin non

Questa scheda appare scegliendo da Repetier Host la voce "Configurazione firmware EEPROM" dal menu Configurazione.

solo gestisce i valori leggendoli dal codice caricato, ma ne fa anche una copia "di lavoro" su EEPROM e – sempre tramite appositi comandi – ne permette la lettura, la modifica, l'uso temporaneo e la memorizzazione permanente su EEPROM.

Il software di stampa Repetier ha un'apposita funzione che permette proprio di gestire i dati in EEPROM del firmware interrogando Marlin e gestendo gli opportuni comandi di lettura e scrittura; ovviamente durante questa operazione la stampante deve essere collegata per leggere i valori. In questo modo si possono fare modifiche, anche persistenti fra un'accensione e l'altra di 3Drag, senza dover riprogrammare la scheda per intero tramite l'IDE di Arduino ad ogni modifica dei parametri.

L'interfaccia si presenta come nell'immagine <REPETIER\_EEPROM> e i dati riportati sono quelli di una nostra stampante all'opera; non tutti i valori sono allineati con quanto indicato sin qui, ma anche noi siamo costantemente in sperimentazione per trovare su lavori diversi le impostazioni

## I comandi per la EEPROM

Ecco i comandi specifici per interagire con i valori contenuti in EEPROM e utilizzati dal firmware per il funzionamento.

- M500** – memorizza i parametri in EEPROM; questi sono:  
axis\_steps\_per\_unit, (M92)  
max\_feedrate, (M203)  
max\_acceleration, (M201)  
acceleration, (M204)

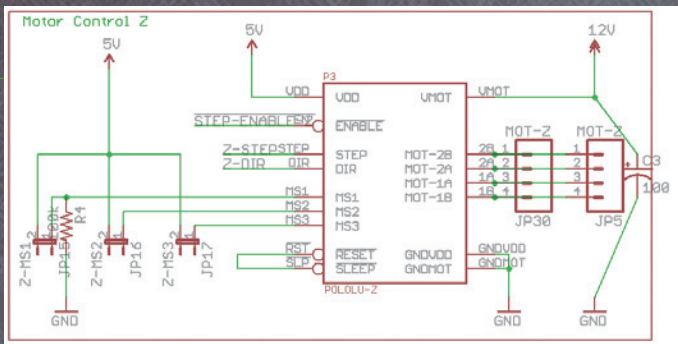
- retract\_acceleration,  
minimumfeedrate, (M205 S T B X Y)  
mintravelfeedrate,  
minsegmenttime,  
jerk velocities,  
PID (M301)

- M501** – legge i parametri dalla EEPROM (ad esempio per tornare alle impostazioni permanenti dopo aver

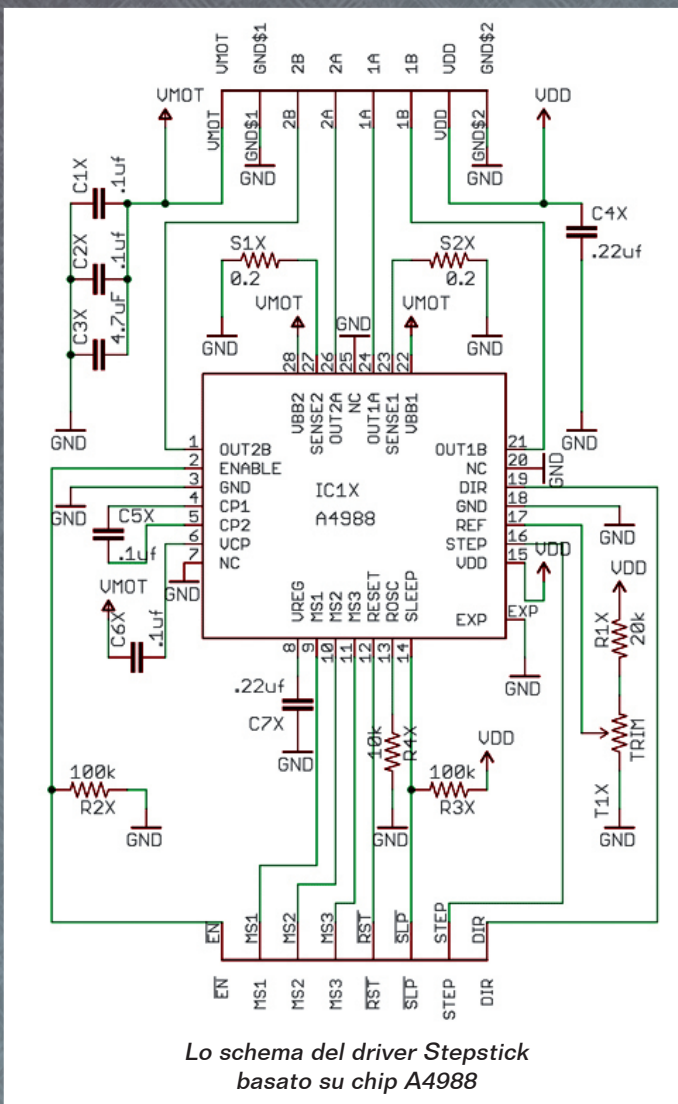
fatto delle modifiche temporanee).

- M502** – torna ai valori di fabbrica, ovvero quelli scritti nel firmware. Vanno memorizzati in EEPROM per usare questi ultimi al posto dei valori memorizzati in precedenza su EEPROM.

- M503** – restituisce i valori correnti presenti in memoria e non in EEPROM.



Lo schema della parte di Sanguinololu che si interfaccia al modulo driver Pololu o Stepstick.



Lo schema del driver Stepstick basato su chip A4988

Table 1. Microstepping Resolution Truth Table

MS1	MS2	MS3	Microstep Resolution	Excitation Mode
L	L	L	Full Step	2 Phase
H	L	L	Half Step	1-2 Phase
L	H	L	Quarter Step	W1-2 Phase
H	H	L	Eighth Step	2W1-2 Phase
H	H	H	Sixteenth Step	4W1-2 Phase

La tabella della verità per la regolazione dei tre jumper sotto a ciascun driver di potenza. I ponticelli sono ordinati da MS1 (a sinistra) a MS3 (a destra) se si tiene il microcontrollore verso di sé.

più adatte. Anche i materiali diversi in fase di estrusione possono richiedere velocità e accelerazioni specifiche, quindi quando una stampa non è soddisfacente, può darsi che sia solo una questione di parametri.

Rispetto a quanto reso accessibile nella schermata, i parametri che possono influenzare la stampa, la qualità e la velocità di stampa e spostamento sono quelli dalla seconda alla quarta riga. Da non toccare invece la prima riga che riguarda il numero di step (microstep) che ciascun motore compie per arrivare a 1 mm. L'unico caso in cui può essere utile cambiare il numero di step è quello relativo a Z: abbiamo scelto di gestire questo asse con 16 microstep, quando in realtà la precisione così ottenuta non ha un riscontro pratico nella qualità della stampa.

### IL DRIVER STEPPER

La scheda Sanguinololu ha sotto ciascun driver (Pololu o StepStick) tre ponticelli che servono a gestire i microstep secondo la tabella della verità riportata nel datasheet (vedi **Tabella 1**).

Osservando lo schema del driver di potenza si può notare come tutti i pin dedicati al pilotaggio della scheda siano da un lato, mentre quelli che vanno agli avvolgimenti dei motori siano dall'altra. Questo deriva da una iniziale impostazione progettuale che vedeva le schedine dei driver a fare da ponte fra la scheda logica e quella con i morsetti per i motori. Oggi la situazione è molto diversa e – con una sola scheda – l'organizzazione di Sanguinololu è quella che vedete, con tutte le connessioni ordinate da un lato grazie alla razionalità di progettazione delle schede driver, rimaste inalterate come formato e piedinatura.

Sempre a proposito di driver, sullo scorso numero abbiamo parlato della taratura della corrente fornita ai motori e della necessità di dotare i chip SMD dei driver di un dissipatore per evitarne il surriscaldamento. In alcune versioni con 4 strati di rame e non solo 2, la dissipazione avviene anche attraverso il circuito stampato. Un piccolo ventilatore, con l'opportuno supporto, può rendere la situazione ottimale e al proposito vi invitiamo a dar libero sfogo alla vostra fantasia per progettare e stampare proprio questa parte, da montare sulla vostra 3Drag.

### IL PIANO DI STAMPA

Dopo aver dedicato molta attenzione alla parte di configurazione, è ora di dedicare un po' di tempo al piano di stampa che, per la sua natura, è respon-



Il piano di stampa in vetronite va periodicamente passato con una spugna abrasiva come questa per esporre la fibra di vetro e permettere la presa del PLA sulla superficie.



sabile di parte del risultato finale. Una buona aderenza, la tenuta anche delle stampe che si estendono in lunghezza e una ragionevole facilità nel rimuovere gli oggetti sono tutte caratteristiche legate al piano di stampa che, come sapete, per 3Drag è in vetronite e non è riscaldato. Per avere sempre alla massima efficienza il vostro piano di stampa, è necessario trattarlo nel modo opportuno a seconda del materiale che intendete stampare. Se pensate di utilizzare il PLA, dovete rendere la superficie leggermente porosa e con le fibre della vetronite esposte. Sembra un concetto complicato, ma basta usare una spugnetta abrasiva, come quella riprodotta in foto e reperibile nei centri di fai da te, per trasformare la superficie da lucida e non adatta all'adesione a opaca e ideale per le stampe in PLA.

Con la spugnetta di tipo medio o fine iniziate a grattare in modo uniforme finché non noterete che si inizia a vedere la trama della vetronite (una serie di aree più chiare e a scacchiera); con un pannello per la polvere rimuovete la polvere creata e a questo punto provate a stampare qualcosa. Noterete che il PLA fuso aderisce alla vetronite sottostante come se la stesse verniciando. Fate anche attenzione alla distanza del piatto dalla testina di stampa: questa deve toccare senza far sforzo su tutti e quattro gli angoli della piastra e anche nel centro.

A fine stampa di prova, con un taglierino a lama larga sollevate l'oggetto stampato partendo da un angolo. Se l'oggetto è sottile sarà più facile, mentre se è spesso o massiccio, dovrete usare più pazienza e lavorare il distacco da più punti.

Una volta che il pezzo inizia a staccarsi, cercate di spingere la lama tenendola il più possibile piatta sul piano di stampa, così da non fare troppo leva su lama e pezzo stampato. Da notare che all'inizio la vetronite tenderà a non far aderire molto il PLA, ma con un po' di uso e qualche passata con la spugna abrasiva, avrete una superficie via via più porosa e di buona tenuta per stampe di oggetti sia larghi che alti. Il PLA, inoltre, è molto meno soggetto al ritiro e al distacco dal piatto di quanto lo sia l'ABS.

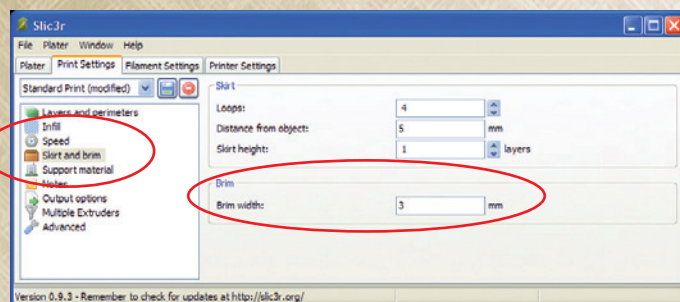
Quando il pezzo inizia a staccarsi con troppa facilità, il piatto ha nuovamente bisogno di un trattamento con la spugna abrasiva, ma potrebbe anche

essere un problema di grasso e sporco: ogni volta che toccate il piatto, lasciate delle impronte che potrebbero anche contenere del grasso cutaneo e quindi rendereste molto più difficile la tenuta del PLA. Se le ditate sono evidenti, utilizzate uno sgrassatore per vetri e un panno: usate poco prodotto e attendete che questo sia completamente evaporato dopo aver passato il panno prima di riprendere le stampe. NON usate prodotti che lasciano residui in quanto i loro effetti sono tutti da scoprire e non è assolutamente detto che siano a favore dell'adesione al piatto del PLA. Nella nostra esperienza, anche la temperatura del primo strato è importante: il PLA, quando è molto liquido, tende ad aderire maggiormente in quanto riempie meglio e si aggrappa alle micro asperità della basetta, mentre se è più viscoso, non penetra così a fondo e anche se lo strato è ben "spalmato", aderisce meno.

### COMBATTERE IL DISTACCO CON IL BRIM

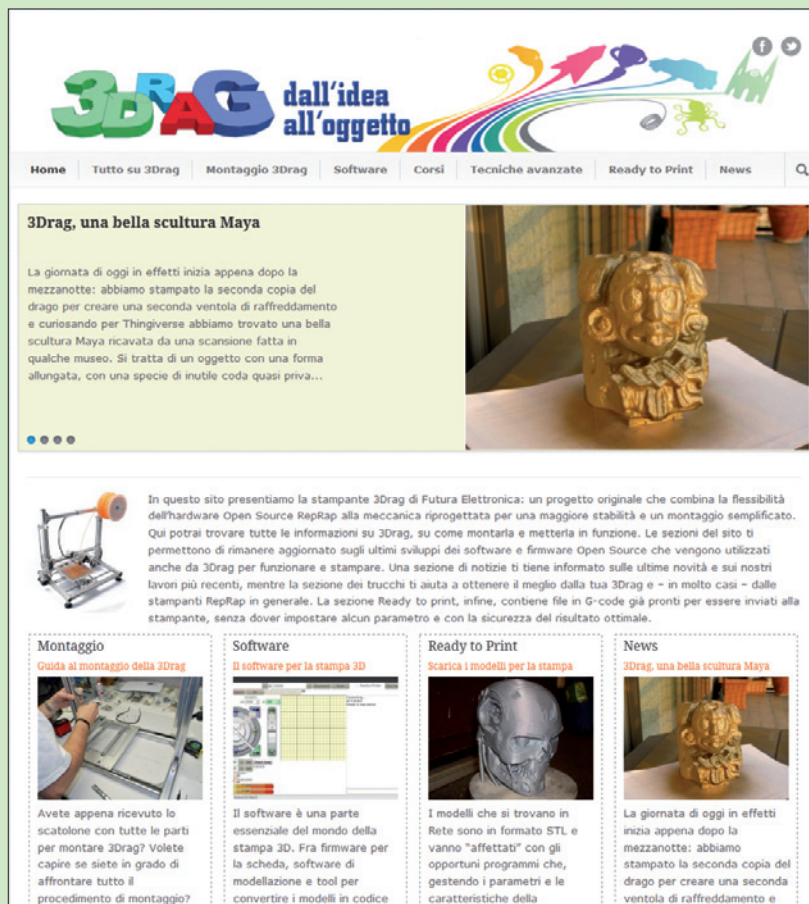
Negli ultimi mesi, dopo aver "pensionato" la soluzione con il Raft, ovvero una superficie creata con un filo appoggiato sul piano di stampa e intrecciata in modo da essere leggera e facile da rimuovere, ma adatta a ridurre le deformazioni da raffreddamento, la comunità degli utenti ha trovato nel "brim" il nuovo alleato per portare a termine le stampe più impegnative e domare la ritrazione termica anche sui piatti non riscaldati.

L'idea di questa funzione è quella di creare una piattaforma che segue tutto il perimetro dell'oggetto esternamente, con una larghezza definita dall'utente. Essendo un solo strato, la ritrazione termica viene controbilanciata e ostacolata dall'ampia superficie di contatto con il piano di stampa. La piattaforma viene generata con abbondante estrusione di materiale proprio per essere certi della presa, quindi il normale perimetro si aggan-



*In evidenza la voce da selezionare a sinistra per avere nella scheda di destra il parametro Brim.*

# Il sito internet di riferimento







**3DRAG** dall'idea all'oggetto

Home Tutto su 3DRAG Montaggio 3DRAG Software Corsi Tecniche avanzate Ready to Print News

### 3DRag, una bella scultura Maya

La giornata di oggi in effetti inizia appena dopo la mezzanotte: abbiamo stampato la seconda copia del drago per creare una seconda ventola di raffreddamento e curiosando per Thingiverse abbiamo trovato una bella scultura Maya ricavata da una scansione fatta in qualche museo. Si tratta di un oggetto con una forma allungata, con una specie di inutile coda quasi priva...

In questo sito presentiamo la stampante 3DRag di Futura Elettronica: un progetto originale che combina la flessibilità dell'hardware Open Source RepRap alla meccanica riprogettata per una maggiore stabilità e un montaggio semplificato. Qui potrai trovare tutte le informazioni su 3DRag, su come montarla e metterla in funzione. Le sezioni del sito ti permettono di rimanere aggiornato sugli ultimi sviluppi dei software e firmware Open Source che vengono utilizzati anche da 3DRag per funzionare e stampare. Una sezione di notizie ti tiene informato sulle ultime novità e sui nostri lavori più recenti, mentre la sezione dei trucchi ti aiuta a ottenere il meglio dalla tua 3DRag e - in molti casi - dalle stampanti RepRap in generale. La sezione Ready to print, infine, contiene file in G-code già pronti per essere inviati alla stampante, senza dover impostare alcun parametro e con la sicurezza del risultato ottimale.

<b>Montaggio</b> Guida al montaggio della 3DRag  Avete appena ricevuto lo scatolone con tutte le parti per montare 3DRag? Volete capire se siete in grado di affrontare tutto il procedimento di montaggio?	<b>Software</b> Il software per la stampa 3D  Il software è una parte essenziale del mondo della stampa 3D. Fra firmware per la scheda, software di modellazione e tool per convertire i modelli in codice	<b>Ready to Print</b> Scarica i modelli per la stampa  I modelli che si trovano in Rete sono in formato STL e vanno "affettati" con gli opportuni programmi che, gestendo i parametri e le caratteristiche della	<b>News</b> 3DRag, una bella scultura Maya  La giornata di oggi in effetti inizia appena dopo la mezzanotte: abbiamo stampato la seconda copia del drago per creare una seconda ventola di raffreddamento e
---	--	--	---

Per supportare quanti hanno acquistato o stanno per acquistare una stampante 3DRag - ma anche per gli utenti della galassia RepRap - abbiamo dato vita ad un sito dedicato solo a questo argomento, ovvero ad un sito nel quale chiunque intenda intraprendere la costruzione della nostra printer 3DRag possa trovare le informazioni indispensabili alla costruzione, alla scelta dei programmi ed alle configurazioni ottimali. Questo l'indirizzo: <http://3drag.elettronica.in.it>

Un sito che nei prossimi mesi, anche con l'aiuto degli utenti, cercheremo di migliorare e ampliare, fornendo anche una serie di modelli pronti all'uso, semplicemente da scaricare, e con le info relative alle configurazioni di stampa ottimali. I modelli che si trovano in Rete, infatti, sono in formato STL e vanno "affettati" con gli opportuni programmi che, gestendo i parametri e le caratteristiche della stampante, creano un file con le istruzioni G-Code che, strato dopo strato, creano l'oggetto finito. Sulla base della nostra esperienza e

delle stampe di prova effettuate con la 3DRag, siamo in grado di offrire dei file G-Code già pronti, realizzati con le migliori impostazioni per ottenere un risultato impeccabile. Usare file G-Code consente anche di iniziare subito la stampa, risparmiando il tempo di slicing, evitando di aver spiacevoli sorprese per qualche parametro non impostato nel modo migliore.

Il sito comprende anche un'ampia sezione dedicata al firmware della printer ed al software per la stampa 3D: fra firmware per la scheda, software di modellazione e tool per convertire i modelli in codice pronto per la stampa, sono molti i fronti da tenere sotto controllo per non perdere novità e aggiornamenti. Questa sezione consente di orientarsi tra versioni, aggiornamenti e derivazioni. Completano il sito una sezione di news, l'area "Tecniche avanzate" e la sezione dei Corsi da noi organizzati in tutta Italia per quanti desiderano, prima dell'acquisto della stampante, conoscere in maniera più approfondita questo mondo.

<http://3dprint.elettronica.in.it>

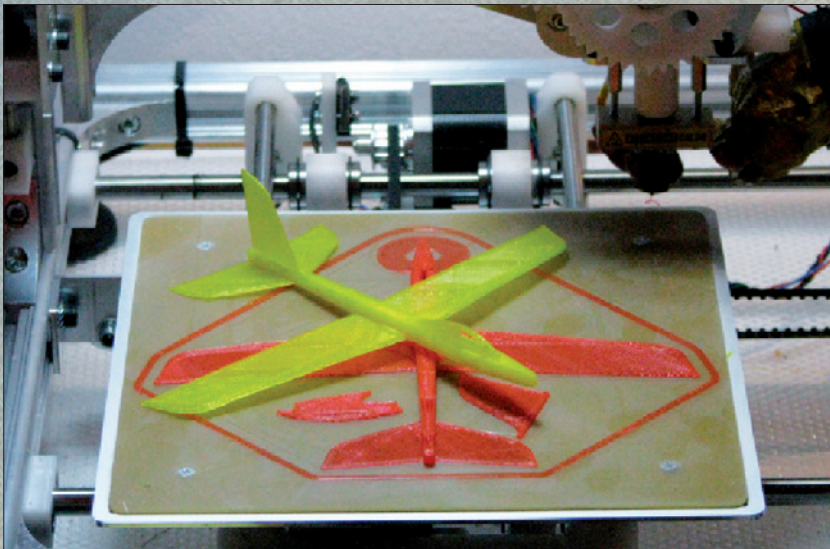
cia ad essa sul bordo esterno. Al termine della stampa il bordo aiuta a staccare con maggiore facilità l'oggetto in quanto la lama si infila sotto al pezzo guidata dalla sottile piattaforma che, grazie alla sua flessibilità è facile da staccare. Con un taglierino si dovrà poi rimuovere questo bordo che, come constaterete, è ben attaccato al resto del pezzo.

Le dimensioni del Brim sono da impostare nel profilo di stampa e sono nella medesima scheda di Slic3r dedicata allo skirt.

A seconda delle dimensioni dell'oggetto e all'intensità della contrazione che vi aspettate per lo specifico pezzo da stampare, potete dare una dimensione più o meno ampia al brim. Ricordate di salvare il profilo con le nuove impostazioni, quindi da Repetier Host, perché queste vengano utilizzate per lo slicing del modello caricato; dovrete prima selezionare un profilo diverso da quello che avete salvato, quindi potete riselezionarlo caricando i nuovi parametri. Se state invece effettuando lo slicing direttamente da Slic3r, i parametri modificati sono immediatamente utilizzati. Vale la pena notare, comunque, che non tutte le stampe hanno bisogno del brim, come non tutti i modelli richiedono i supporti. Nell'immagine dell'aeroplanino che trovate in queste pagine, l'uso del brim non darebbe alcun vantaggio, ma addirittura comprometterebbe il risultato finale con un oggetto da ripulire con non poche difficoltà a causa della sua struttura fatta da pochissimi strati.

## AGGIUNGERE UNO STRATO ADESIVO

Ci sono casi in cui la stampa richiede una presa molto salda sul piano di stampa, come ad esempio per i vasi che hanno



*Questo modello di aeroplano è fatto da due soli strati per le ali, con dei bordi di irrigidimento e un corpo alto circa 4 mm. Per sua natura aderisce perfettamente al piano di stampa e non necessita di "brim".*

una base molto piccola e un certo peso da sopportare per un lungo periodo di spostamenti e vibrazioni, oppure soggetti che hanno un piedistallo e poi si sviluppano in altezza come i busti e le statuette. Per questi, il tipo di presa che la sola vetronite offre non è sufficiente e per questo ci sono soluzioni alternative. La più comune si basa sulla colla vinilica, nota con il nome commerciale Vinavil. Questa colla, se diluita con acqua in parti uguali (1 di colla e 1 di acqua), può essere stesa sul piatto di stampa spennellando come se si dovesse semplicemente bagnare tutta la superficie. Quando lo strato è uniforme, si deve lasciare asciugare il tutto con un po' di pazienza. La colla al tatto deve risultare liscia e senza punti appiccicosi. Solo quando si è sicuri di questo si può tornare a stampare. A differenza della sola vetronite, lo strato di colla risulta vulnerabile ai passaggi troppo bassi dell'ugello e in casi sfortunati può addirittura essere rimossa dall'ugello che lascia un solco ben visibile. Per questo motivo, sia la taratura dello "0" sull'asse

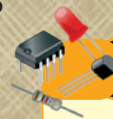
Z, sia l'omogeneità dello strato di colla vanno verificati con attenzione. Lo strato di colla, comunque, potrà essere rimosso in ogni momento con il taglierino o lavando con acqua calda il piano di stampa (staccatelo dalla stampante prima di lavarlo). Tenete presente che la presa della colla sul PLA può essere molto forte e dipende dalla temperatura a cui stampate: se PLA e colla si fondono troppo fra loro, il distacco potrebbe diventare molto problematico. Se usate lo strato di colla, potete generalmente fare a meno del Brim, anche se quest'ultimo vi può aiutare a guidare sotto il pezzo la lama del taglierino. Rispetto al solo piano in vetronite, la colla tende a rendere meno liscia e uniforme la superficie inferiore di quanto stampate. Con le varie stampe, vi capiterà anche di staccare in alcuni punti la colla, lasciando scoperta la vetronite. Le parti rimaste di colla garantiranno la presa, ma quando sarà rimasta poca colla o avrete fatto stampe ripetitive che hanno di fatto "ripulito" le aree di contatto a furia di ripassarci sopra, dovrete asportare i rima-



*Il classico flacone bianco e rosso di Vinavil: la colla che permette di aumentare molto la presa sul piatto di stampa.*

sugli e fare una nuova stesura di colla diluita.

Nel prossimo numero ci dedicheremo all'esplorazione delle varie funzioni del programma Slic3r, di fatto il principale responsabile della qualità e del successo delle vostre stampe. ■



## per il MATERIALE

La stampante 3Drag è disponibile sia in scatola di montaggio (3DRAG/K, Euro 720,00 +IVA) che già montata e collaudata (3DRAG/M, Euro 910,00 + IVA). Il software necessario al funzionamento della stampante è completamente open-source ed è scaricabile gratuitamente da Internet. Presso Futura Elettronica sono disponibili anche i fili in ABS e PLA, in differenti formati e colori, necessari per realizzare le stampe. Per questo prodotto esiste anche un sito di riferimento: <http://3dprint.elettronica.in>

Il materiale va richiesto a:  
Futura Elettronica, Via Adige 11,  
21013 Gallarate (VA)  
Tel: 0331-799775 - Fax: 0331-792287  
- [www.futurashop.it](http://www.futurashop.it)