

# Soldering Machine Handbook

James Glanville

15th May 2013

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Prerequisites</b>	<b>2</b>
<b>3</b>	<b>Installing Software</b>	<b>2</b>
3.1	Project files . . . . .	2
3.2	Install pronterface (if PC control is required) . . . . .	2
3.3	Preparing SD card . . . . .	2
3.4	Copying utility GCODE to SD card. . . . .	3
<b>4</b>	<b>Creating GERBER files.</b>	<b>3</b>
<b>5</b>	<b>Generating GCODE with pcb2gcode</b>	<b>3</b>
5.1	Creating millproject file. . . . .	3
5.2	Copy GCODE to SD Card . . . . .	3
<b>6</b>	<b>Operating the machine</b>	<b>3</b>
6.1	Attaching the PCB . . . . .	3
6.2	Isolation Routing . . . . .	3
6.3	Drilling . . . . .	4
6.4	Paste Extrusion . . . . .	4
6.5	Placing components . . . . .	4
6.6	Reflowing the board . . . . .	4
	<b>Appendices</b>	<b>6</b>
<b>A</b>	<b>Building pcb2gcode from source</b>	<b>6</b>
A.1	Compilation in Linux . . . . .	6
A.2	Compilation in Windows . . . . .	6
A.2.1	Visual Studio . . . . .	6
A.2.2	GTK+ development files . . . . .	6
A.2.3	Gerbv . . . . .	7
A.2.4	Boost libraries . . . . .	7
A.2.5	GTKMM . . . . .	7
A.2.6	Gerbv source . . . . .	7
A.2.7	Compilation . . . . .	7

# 1 Introduction

This handbook lists the steps required to operate the machine, and produce populated PCBs. The operating system used in this handbook is Ubuntu 13.10. The PCB CAD program used is Kicad.

## 2 Prerequisites

The following things are assumed:

- Computer with Windows/Linux. OS X may work, but is untested.
- SD card (at least 128mb).

## 3 Installing Software

First install pcb2gcode as described in [Appendix A](#).

### 3.1 Project files

Fetch the project's files from source control:

```
git clone https://github.com/JamesGlanville/solderingmachine.git
```

### 3.2 Install pronterface (if PC control is required)

```
git clone https://github.com/kliment/Printrun.git
```

### 3.3 Preparing SD card

Format the SD card to the FAT32 filesystem, ensuring the disk has an MSDOS partition table and that the FAT32 partition is on the first primary partition.

In Ubuntu, the following steps are necessary:

Install gparted if it is not already installed:

```
sudo apt-get install gparted
```

Launch gparted as root:

```
sudo gparted
```

Select the SD card using the drop down menu at the top right of the screen.

Create the MSDOS partition table by clicking Device->Create Partition Table.

Create the FAT32 filesystem by right-clicking on the empty disk, and selecting "New". Fill in the information in the dialogue box, and click "Apply changes".

### 3.4 Copying utility GCODE to SD card.

Copy all .GCODE files from pcb2gcode-metric/GCODE/ to the root of the SD card. These can be used to perform simple functions such as zeroing the axes.

## 4 Creating GERBER files.

The following steps should be followed to export GERBER files from Kicad:

1. Open the PCB layout in PCBNew.
2. Select **Tracks and Vias** from the **Dimensions** Menu.
3. Change the **Mask Clearance** to 0.0001.
4. Select **Plot** from the **File** menu.
5. Select the following layers: **Copper, Component, SolderMask, EdgesPcb**
6. Click **Plot** and then **Generate drill file**.

## 5 Generating GCODE with pcb2gcode

### 5.1 Creating millproject file.

An example millproject file is given in pcb2gcode-metric/millproject. This should be copied into the directory containing the GERBER files, and modified if necessary (see included comments).

### 5.2 Copy GCODE to SD Card

## 6 Operating the machine

The following steps described the various stages involved in producing a populated and soldered circuit board. Some steps (such as drilling) may be omitted if not required.

### 6.1 Attaching the PCB

Attach copper-clad PCB to machine using double sided tape. It is important that the PCB is level, so care must be taken ensure the tape is not bunched or folded. The PCB should be firmly pressed to the bed to ensure a strong bond.

### 6.2 Isolation Routing

1. Insert conical routing bit into the spindle. This is achieved by loosening the idler section, and then loosening the two machine screws holding the bit in place. The bit should be carefully removed by hand, and the conical routing bit pushed into its place. The two machine screws should be tightened, and the idler tightened.

2. The spindle tool should then be attached to the tool holder with the two wing nuts.
3. Switch on the machine.
4. Insert the SD card containing all of the GCODE files and power on the machine.
5. Run **zero.gcode** then **startspindle.gcode** then **mill.gcode**
6. When routing is complete, run **stopspindle.gcode**

### 6.3 Drilling

This section may be ignored if no holes need be drilled through the PCB.

1. The drill bit should be inserted into the spindle tool as described in the "Isolation Routing" section above.
2. Switch on the machine.
3. Insert the SD card containing all of the GCODE files and power on the machine.
4. Run **zero.GCODE** then **startspindle.GCODE** then **drill.GCODE**
5. When drilling is complete, run **stopspindle.GCODE**

### 6.4 Paste Extrusion

1. Insert a syringe containing a sufficient volume of solder paste into the paste extruder. Hand-tighten the extruder so that there is only a small amount of slack in the timing belt.
2. Attach the paste extruder to the tool holder and hand-tighten the wing nuts.
3. Switch on the machine.
4. Insert the SD card containing all of the GCODE files and power on the machine.
5. Run **zero.GCODE** then **prime.GCODE** then **paste.GCODE**

### 6.5 Placing components

1. Attach the vacuum needle tool to the machine and tighten with the wing nuts.
2. Fill the part tray with the necessary SMD components.
3. Switch on the machine.
4. Insert the SD card containing all of the GCODE files and power on the machine.
5. Run **zero.GCODE** then **place.GCODE**.

### 6.6 Reflowing the board

1. Attach the hot air tool to the machine and tighten with the wing nuts.
2. Switch on the machine.

3. Insert the SD card containing all of the GCODE files and power on the machine.
4. Run **zero.GCODE** then **reflow.GCODE**.
5. Run **cooldown.GCODE**

# Appendices

## A Building pcb2gcode from source

The pcb2gcode source should first be obtained:

```
git clone git@github.com:JamesGlanville/pcb2gcode-metric.git
```

### A.1 Compilation in Linux

For these instructions the use of Ubuntu/Debian is assumed, although other distributions should be similar.

To install prerequisites:

```
sudo apt-get install build-essential automake autoconf libtool libboost-all  
-dev libgtkmm-2.4-dev gerbv checkinstall
```

To compile pcb2gcode:

```
cd pcb2gcode-metric  
./git-build.sh
```

(Optional) To install pcb2gcode:

```
sudo checkinstall
```

### A.2 Compilation in Windows

Installing prerequisites is more complex in Windows:

#### A.2.1 Visual Studio

Install Visual Studio 2012 Professional from:

```
www.visualstudio.com
```

#### A.2.2 GTK+ development files

Download:

```
http://ftp.gnome.org/pub/gnome/binaries/win32/gtk+/2.24/gtk+-bundle_2  
.24.10-20120208_win32.zip
```

and extract to pcb2gcode-metric folder.

### A.2.3 Gerbv

Download and install gerbvinst-2.6.1.exe from:

```
http://sourceforge.net/projects/gerbv/
```

### A.2.4 Boost libraries

Download boost\_1\_55\_0.zip from:

```
http://sourceforge.net/projects/boost/
```

Unzip contents to c:\boost155 open VS2012 x86 Native Tools Command Prompt, from:

```
C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Microsoft Visual  
Studio 2012\Visual Studio Tools
```

As administrator, run the following commands:

```
cd c:\boost155  
bootstrap.bat  
bjam.exe
```

### A.2.5 GTKMM

Download and install gtkmm-win32-devel-2.22.0-2.exe from:

```
http://ftp.gnome.org/pub/GNOME/binaries/win32/gtkmm/2.22/
```

### A.2.6 Gerbv source

Download gerbv-2.6.1.tar.gz from:

```
http://sourceforge.net/projects/gerbv/files/gerbv/gerbv-2.6.1/
```

and extract to c:\gerbv-2.6.1

### A.2.7 Compilation

The file pcb2gcode-metric.sln may now be opened in visual studio and the project can be compiled.