CNC Milling 101

In this course at TTLMakerspace, students will go from beginner to a practical working knowledge of CAD modelling and CNC machining with an automated milling machine.

Starting with desktop versions of CNC machines, participants work their way through lessons up to the industrial Omni 1350 6' x 8' CNC milling machine.

Week 1
Introduction to concepts of CNC.

This week students get a very good introduction to the basic principles of CNC, including learning about each axis, gcode, and the full workflow of producing a part from a rough idea. They will practice sending gcode manually to their mill, and learn the significance of the most important codes.

3 Axis
Gcode
Gcode sender
Role of a CAD program
Role of a CAM program
Special considerations of milling
3018 CNC Overview

Genmitsu Specifics:

- 775 Motor at 12 volts,
- max rpm 10,000 (this is quite tiny)
- NEMA 17 stepper motors at 1.7amps max, 9-42 volts

Exercise: install everything on linux for a 3018 CNC

- Download and run ugs, learn to use a shell script on linux if you have never done that before

When you first plug in your 3018, there is a good chance that it will not be seen. This is because we need to install drivers. A driver is a small application that recognizes the signature of the device and creates an API for communication between your computer and the device.

In the case of a GRBL based 3018 CNC (such as the woodpecker) the driver you need to install is: CH340G USB chipset

Bad news if you're on a mac - no drivers exist for MAC OS for this chipset!

The arduino IDE is a good way to do basic communication between your device and the computer, including checking configuration files via the serial port (USB)

Install the Arduino IDE on linux. When first starting Arduino, the IDE may tell you that you do not have permissions to access the USB with the current user. This is a protection of the linux OS, and it is simple to address.
Use ‘dmesg’ - this is your tool to see what is happening with your USB ports. Try running it and plugging/unplugging your usb devices, including the CNC. You’ll see that the drivers are there, so chances are you just need the permissions.

Once you have set the permission, you’ll be able to access the USB. (you need to reboot to enable this)

Once rebooted, you’ll be able to communicate with the grbl board via Arduino. Start arduino. Now, when you click on Tools>port you’ll see the device port your GRBL is connected to, select it. Now open the serial monitor.

Talking over serial port can be done at different speeds, since some devices have faster chipsets than others. Both devices have to speak at the same speed to get communication working. For the CNC the speed is 11500 - select this in the “baud” area of the serial port monitor.

You have communication, and you’ll see the version of grbl installed!

Type $$ into the serial port, you’ll see a bunch of codes come up. This is the configuration of your grbl board, and these determine things like whether limit switches are installed, etc. These settings can be updated through a serial terminal such as arduino. This is very useful for confirming your board is set up correctly, and for modding your board (for instance if you upgrade to a higher power spindle).

GRBL Configuration Values

Now we can try out UGS. To do it, we’ll need to close out our serial terminal, since only one software can talk to the device through the serial port at a time.

Launch UGS through the shell script. Refresh the port selector. Select the same port you accessed using Arduino IDE.

Notice that the console works just like your Arduino serial terminal!

Try typing $$ into it.

Notice that you get back the same codes, but your console supplies a bit more information along with it. This is UGS recognizing those codes, and automatically filling in more information that makes it more human-readable.

How To Read GCode

These Gcodes are not specific to grbl, they are for ALL control boards, which makes Gcode very versatile for controlling a wide variety of machines.

Most important codes:
G21 - millimeters
G20 - inches
G00 - use rapid positioning (not cutting)
G01 - linear motion
G02 - circular motion clockwise
G03 - circular motion counterclockwise
G17/18/19 - select working plane (x/y, x/z, y/z)
G28 - return home
G90 - absolute mode
G91 - relative mode
M00 - stop program
M02 - end of program
M03 - activate spindle clockwise
M04 - activate spindle counterclockwise
M05 - spindle stop
M06 - tool change

In this first lesson, we will provide an introduction to the OMNI CNC so that we can compare the differences. Students will be made familiar with the way the A11 controller works, and will be given tasks to achieve by using the keypad. They will not get into making cuts this week.

RichAuto A11 USB Control

Omni 1350 Specifics

- X,Y,Z working area: 1300*2500*200mm
- Table size: 1500*3040mm
- Spindle: 3kw watercooling spindle
- Motor : stepper, Yako
- Vacuum System: vacuum table
- X, Y, Z Travelling Positioning Accuracy 0.05/300mm
- X, Y, Z Travelling Repositioning Accuracy 0.05mm
- Frame: Steel Square Tubes
- X, Y structure: rack and pinion drive, hiwin rail linear bearing

- Max Rapid Travel Rate: 50,000mm/min
- Max working speed: 25,000mm/min
- Working voltage: AC380V/50Hz
- Command languages: G code, *mmg, *plt, *u00

The Omni 1350 CNC router differs from the Genmitsu CNC in a few ways.

In terms of electronics, the control board is a DSP controller compared to the GRBL control board of the smaller Genmitsu.

The board is accessed via the A11 offline controller - a small keyboard/lcd combination that uses a 50 pin data transmission cable, compared to the USB serial cable of the GRBL controller.

It is not possible to stream G-code to the controller board directly from your computer using a G-code sender. Instead, files have to be loaded into the A11 either via USB key, or using a USB cable and accessing the A11 like a flash drive.

Once the file is on the A11, the menu system is used to send the file to the DSP controller and start the operation.

The disadvantage of this is that there is virtually no feedback. If the Omni hits a problem, an error code may be shown on the LCD screen, but it will not be possible to examine the G code stream to really know what happened or how the problem occurred.

The advantage of the A11 is mostly industrial - you can load it up with the specific files you need to produce, then just fire them through to the router with no need to set up a computer or install software.

Given the use of the offline controller, it is important to memorize the options of the menu system, as you will not get software help while using it. The embedded software is called the RichAuto system, and the important commands are input by either singe, or double key presses on the built in keypad. In he case of double-presses, you will hold down the control key, then press down on the action key, then release both at the same time.

The RichAuto software define 16 operation keys. Each key has one or more functions depending on the state of the machine, which can be put into several different modes.

Two Key Examples:

“MENU” + “9”	- Set to work coordinate system
“MENU” + “0”	- Set to mechanical coordinate system

This changes configuration between MCS and WCS. In MCS mode, the machine will base all operations from the zero point defined by the actual hardware. In WCS mode, the machine will use software-defined zero coordinates that correspond to the work material added to the bed. For the most part WCS is the preferred mode.

“MENU” + “ON/OFF” - Start the Z-axis automatic tool setting

This command will start the spindle descending slowly toward a special metal “puck” used to calibrate the z-depth. When the electrically conductive spindle (and tool) contact the puck a current is triggered which makes the software stop the spindle’s movement and records the calibration of the z-depth.

One Key Examples:

Most keys have more than one use, and the use will be determined by the mode the machine is in.

Consider the key for “X+” “1” “^”. This is a single key with three separate uses.

“X+”	- Move spindle in x axis positive direction (in movement mode)
“^”	- Menu upwards (in menu mode)
“1”	- Enter a one (in entry mode)

In addition to these three modes, there is also the “processing” mode that occurs while the machine is actively operating. Some keys can affect the machine during processing.

“Y+”	- Move spindle in y axis positive direction (in movement mode)
	- Accelerate movement speed (in process mode)
“^”	- Select another property (in menu mode)
“2”	- Enter a two (in entry mode)

Exercises:

The OMNI can be made to move in either manual or automatic modes. Manual mode is controlled by key presses, and there are three options for how far/fast the gantry should move.

Manual Mode Processing

Continuous Mode
Step Mode
Distance Mode

Automatic Mode Process

This requires a gcode file to be loaded into memory. Automatic mode is what is used when entering into full operation of the machine. To use automatic processing mode, a few steps must be taken initially.

- Set the workpiece origin, including the Z-depth
- Select the work file from memory/disk
- Set processing parameters (processing speed, travel speed, Z down ratio, speed ratio, spindle grad, pulse equivalent, and Z up distance)

The system will perform a quick check of the file, after which it will commence operation.

During Processing

While the operation is in process, it is possible to manually adjust some settings, such as travel speed, and spindle speed. It is also possible to pause the operation, which will stop all x,y,z travel but will leave the spindle running.

Travel speed changes are done by applying a ratio to the base speed. The ratio can be set from 1.0 to 0.1

“Y+” -adjust travel (x,y) speed ratio upwards
“Y-” -adjust travel (x,y) speed ratio downwards

Spindle changes are affected by grade. Each increase/decrease changes the spindle grade by one, to a maximum of 8 grades.

“Z+” -adjust spindle grade upwards by one
“Z-” -adjust spindle grade downwards by one

Entering Paused Mode

“RUN/PAUSE”-enter paused mode

While paused, it is possible to make adjustments to the processing of the current file. At this time the spindle will continue running and it is possible to move the gantry using the manual-mode controls. By default the movement mode will be set to “step” for slow, detailed control. This can be changed to continuous mode for larger movements.

“HIGH/LOW” -change between step and continuous movement

When exiting paused mode, the machine can either return to the pre-paused position and resume operation, or continue operation from the current adjusted position.

“ORIGIN” -return to pre-paused position and resume

“STOP” -resume in current adjusted position

Exercise:
Start a processing operation. Enter paused mode and move the gantry out of the way to do an adjustment to the work material. Exit paused mode so that the gantry/spindle resumes operation of the processing.

Stopping and Break Points

It is possible to entirely stop processing the current file, and when we do so we have the option of saving the current position in a break-point. Up to 8 break points can be stored in memory on the Omni CNC.

“STOP” - stop the current processing operation

The system is now in stopped mode, and prompts to save a break point.

“ORIGIN” - start save break point (choose point by “X+” or “X-”)
“ORIGIN” - complete save break point at position

With the break point saved, you can now choose to exit or resume processing at the break point.

“RUN” + 1-8 - resume processing at the chosen break point

The system can also recover from power failure by saving the current position, spindle speed, and line of g-code the processing of the file has reached. Upon restoration of power the A11 readout will prompt to resume operation.

Exercise:
Stop a processing operation. Save a breakpoint in position 1. Resume processing operation at the breakpoint in position 1. Advanced: change the travel/spindle speed before resuming

This completes basic operation of the Omni. In a later lesson we will look at options for advanced processing of gcode files, including arrays and tool change operations. Advanced processing allows us to resume from breakpoints with different settings, queue up multiple jobs, change tools in the middle of an operation, process partial gcode files, and more.