Contents

Overview and Features

SimplePnP is a Cartesian pick and place machine. It can be divided in two parts: the head and the table.

The head is the part used to grab electronic components meant to be installed on a PCB. It uses a hollow shaft stepper motor to rotate the parts and allow a DC vacuum pump to hold them with different sized nozzles and a camera to help locating features and components.

The table is meant to position the head in the correct spot to hold and release each component. It is a gantry style system built from aluminum extrusion and uses linear rails, stepper motors and timing belts to perform motion.

A computer running OpenPnP is used as the brains of the machine, which sends the appropriate commands to a pair of GRBL motion controllers that handle the operations needed to perform the PCB assembly job.

Its specs are as follows:

X travel	300 mm	
Y travel	300 mm	
Z travel	23 mm	
Footprint	650×500×200 mm	
CPH	750	With vision assist
Component size	Up to 0402 (1005 metric)	
Nozzle type	Samsung CP45	
Vision	Top vision	Optional bottom vision
Number of Heads	1 (base model)	Optional dual head
Power consumption	60 W	With a 12 V power supply
Weight	4 kg	Bench mountable

GRBL

This machine uses a pair of Arduino based GRBL controllers with V1.1 firmware for both the Cartesian table and vacuum head. Only the controller parameters will be presented in this manual. Instructions on how to configure GRBL are available on the web.

Parameters

Table	Head	Table	Head
$0=10	$0=10	$26=250	$26=250
$1=50	$1=255	$27=1.000	$27=1.000
$2=0	$2=0	$30=1000	$30=1000
$3=0	$3=1	$31=0	$31=0
$4=0	$4=0	$32=0	$32=0
$5=1	$5=1	$100=50.000	$100=50.000
$6=0	$6=0	$101=50.000	$101=50.000
$10=1	$10=1	$102=50.000	$102=4.444
$11=0.010	$11=0.010	$110=20000.000	$110=18000.000
$12=0.002	$12=0.002	$111=20000.000	$111=18000.000
$13=0	$13=0	$112=20000.000	$112=18000.000
$20=1	$20=0	$120=1000.000	$120=1000.000
$21=0	$21=0	$121=1000.000	$121=1000.000
$22=1	$22=1	$122=1000.000	$122=1000.000
$23=7	$23=0	$130=310.000	$130=23.000
$24=25.000	$24=25.000	$131=310.000	$131=23.000
$25=1000.000	$25=1000.000	$132=310.000	$132=360.000

OpenPnP

OpenPnP is a Java based open source software program that can be used with this machine in order to program and run PCB assembly jobs. This section of the manual will cover the configuration of machine drivers, head and feeders. In depth information about configuring OpenPnP can be found at https://github.com/openpnp/openpnp/wiki

Machine Drivers

A machine driver contains all the info pertaining the communication between a motion controller (GRBL in this case) and a computer running OpenPnP. SimplePnP uses two separate motion controllers, one for the table and another one for the head, so we need to set up the table driver and the head sub driver.

Table Driver Setup

Setting	Message
COMMAND_CONFIRM_REGEX	OK
CONNECT_COMMAND	$X G90
HOME_COMMAND	$H G92 X0 Y0 Z0
MOVE_TO_COMMAND	G1 {Y:X%.4f} {Y:Y%.4f} {X:Z%.4f} F{FeedRate:%.0f} G4 P0
PICK_COMMAND	M8
PLACE_COMMAND	M9

Head Sub Driver

Setting	Message
COMMAND_CONFIRM_REGEX	OK
CONNECT_COMMAND	$X G90
HOME_COMMAND	G92 X0 Y0 Z0
MOVE_TO_COMMAND	G1 {Z:X%.4f} {Rotation:Z%.4f} F{FeedRate:%.0f} G4 P0

The machine.xml sub driver section should look like this:

```
<sub-drivers class="java.util.ArrayList">
  <gcode-driver communications="serial" connection-keep-alive="false" units="Millimeters"
  rate="18000" backlash-offset-x="-1.0" backlash-offset-y="-1.0" non-squareness-factor="0.0"
  factor="0.1" timeout-milliseconds="5000" connect-wait-time-milliseconds="3000"
  enabled="true" name="GcodeDriver">
    <serial line-ending-type="LF" port-name="COM5" baud="115200" flow-control="Off" data-bits="Eight"
    bits="One" parity="None" set-dtr="false" set-rts="false" name="SerialPortCommunications"/>
    <tcp line-ending-type="LF" ip-address="127.0.0.1" port="23" name="TcpCommunications"/>
    <homing-fiducial-location units="Millimeters" x="0.0" y="0.0" z="0.0" rotation="0.0"/>
    <command type="COMMAND_CONFIRM_REGEX">
      <text><![CDATA[ok]]></text>
    </command>
    <command type="MOVE_TO_COMMAND">
      <text><![CDATA[G1 {Z:X%.4f} {Rotation:Z%.4f} F{FeedRate:%.0f}]]></text>
      <text><![CDATA[G4 P0]]></text>
    </command>
    <command type="CONNECT_COMMAND">
      <text><![CDATA[$X]]></text>
      <text><![CDATA[G90]]></text>
    </command>
    <command type="HOME_COMMAND">
      <text><![CDATA[G92 X0 Y0 Z0]]></text>
    </command>
    <sub-drivers class="java.util.ArrayList"/>
    <axes class="java.util.ArrayList">
      <axis name="z" type="Z" home-coordinate="0.0">
        <head-mountable-ids class="java.util.HashSet">
          <string>  </string>
        </head-mountable-ids>
      </axis>
      <axis name="rotation" type="Rotation" home-coordinate="0.0">
        <head-mountable-ids class="java.util.HashSet">
          <string>  </string>
        </head-mountable-ids>
      </axis>
    </axes>
  </gcode-driver>
</sub-drivers>
```

Head

The head machine component has the information regarding the nozzle carriage and bottom looking camera parameters and capabilities. OpenPnP allows the user to set the preferred head parking location and it is recommended to set the soft travel limits to:

	X	Y
Minimum	0	0
Maximum	300	300

Nozzle

To set up a nozzle for the SimplePnP machine, the user needs to do the following:

- Set nozzle X offset to -30
- Activate “Limit Rotation to 180°”
- Set pick and place dwell times to 350 ms
- Create default nozzle tip and mark it as compatible with the nozzle

Camera

To set up a nozzle for the SimplePnP machine, the user needs to do the following:

- Set top camera as down looking
- Configure units per pixel depending on Z distance from surface
- Calibrate test using built in routine

Top Camera Characteristics

- 2.8 mm focal length
- Manual focus
- M12 threaded lens
- USB 2.0 interface
- 1280x720

Pump specs

- 12 V 10W brushed DC motor
- 10 liters per minute
- 55 kPa max vacuum pressure