

gizDuino

Installation Guide

For gizDuino 168/328 Ver 3.0

gizDuino platform is based on open source Arduino platform designed by Massimo Banzi, et al. It is not an exact copy though, but the user will not notice any difference during normal use. Once the driver is successfully installed, the gizDuino will function exactly like an Arduino. The important thing is, the right driver must be installed in the right manner. Failure to do this simple procedure would almost certainly result in problems.

USB Interface

The Arduino uses an FT-232 chip as its USB bridge. In contrast, the gizDuino uses the PL-2303 chip. PL-2303 was picked over the FT-232 since they are easier to source, at least from our region. This difference will be visible to the user only during device driver installation.

Windows user will notice that the first time the gizDuino is plugged in a PC, the Windows OS will automatically make an attempt to install its own driver. And will usually complete the procedure without flagging an error. The problem is, gizDuino will not work with the Windows built-in driver. You must install the latest the driver provided with the gizDuino.



Figure 1. gizDuino 168/328 Ver. 3.0 now features automatic power source selection and an optiboot bootloader.



Figure 2. Previous versions of the gizDuino require the user to manually change jumper settings to switch to a different power source.

Copyright 2012 by e-Gizmo Mechatronix Central.

Arduino is an open source platform developed by the Arduino Team
(Dave Mellis, Tom Igoe, Gianluca Martino,
David Cuartielles, and Massimo Banzi) .

www.arduino.cc

Driver Installation

Driver installation must be done before connecting the gizDuino with your PC. But then, we have this irresistible urge of plugging things first before reading the manual. You already plugged your gizDuino, didn't you? Not to worry, it is nothing to be embarrassed about. No permanent harm done either, it is just you burdened yourself with the extra work needed to undo your mistake – you have to uninstall the wrong driver. It is easy, just uninstall the PL-2303 com port appearing in your system device driver list. If you have no idea how to do this, have someone to do it for you.

To install, load the installation CD and find the folder with a PL2303 on its label, and run the setup.exe contained in it. Follow the setup instructions that flashes on your screen, and you are done.

COM port Assignment

Don't plug your gizDuino just yet, we have to determine what COM port it is assigned to:

1. Launch the Arduino IDE.
2. Click Tools>Serial Port and record the list of available COM ports. (See Note 1)
3. Click anywhere within the Arduino IDE window to close the drop menu.
4. Plug the gizDuino. Let the PC do its job of recognizing the gizDuino. This may take a few seconds.
5. Click Tools>Serial Port once again and look for a new COM port added to the list. This is the COM assignment of your gizDuino. Click to select if it is not marked with a check mark.

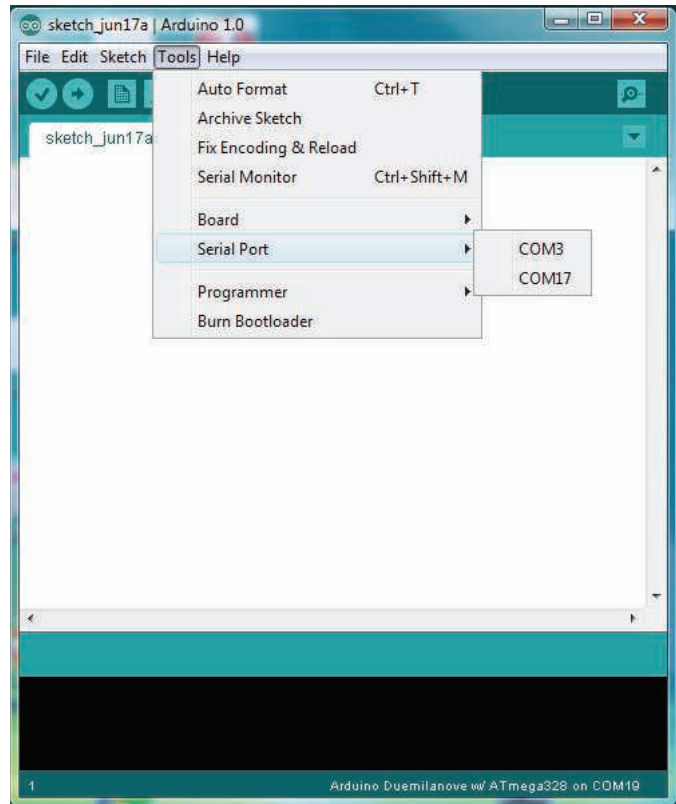


Figure 3. Available COM ports as displayed from my PC before the gizDuino is plugged in. See Note 1.

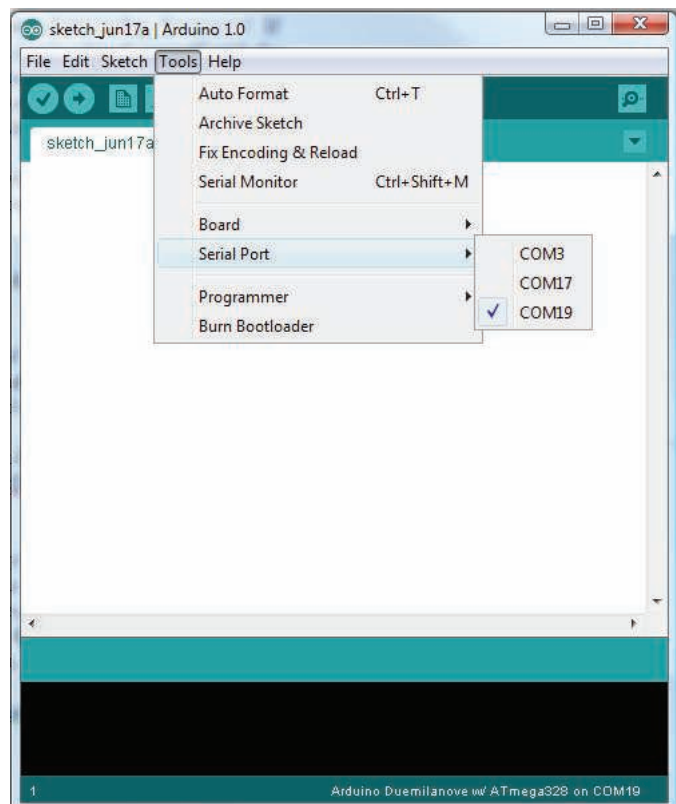


Figure 4. COM19 appears after plugging in my gizDuino. A different COM port may be assigned by your PC depending on its availability.

Board Assignment

We are almost there. The final step will be to tell the Arduino IDE what Arduino hardware you are using. If you are using an Arduino IDE provided by e-Gizmo, just select the gizduino model corresponding to your board. If you acquired it elsewhere, don't look for gizduino boards in the list, there is none. Select the equivalent as follows:

gizduino 328 rev 3.0 -> Arduino Uno

If you are using an older version of gizduino:

gizduino 328

-> Arduino Duemilanove w/ ATmega328

gizduino 168

-> Arduino Diecimila or Duemilanove w/ ATmega168

Your gizduino is now ready to use! This procedure need to be done only once. Once the setup is correctly completed, your gizduino will be remembered by the Arduino IDE for as long as you want it to be remembered.

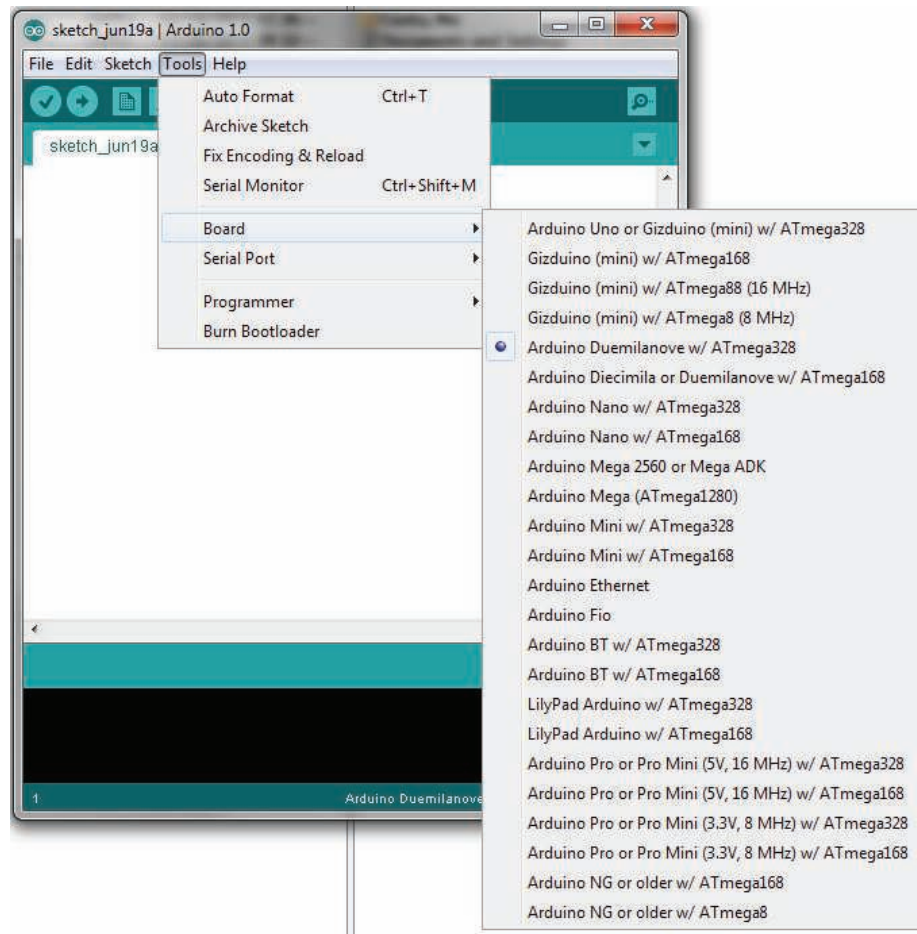


Figure 5. Select the gizduino model corresponding to your board. If the gizduino boards do not appear in the list, select the Arduino equivalent as described in the text.

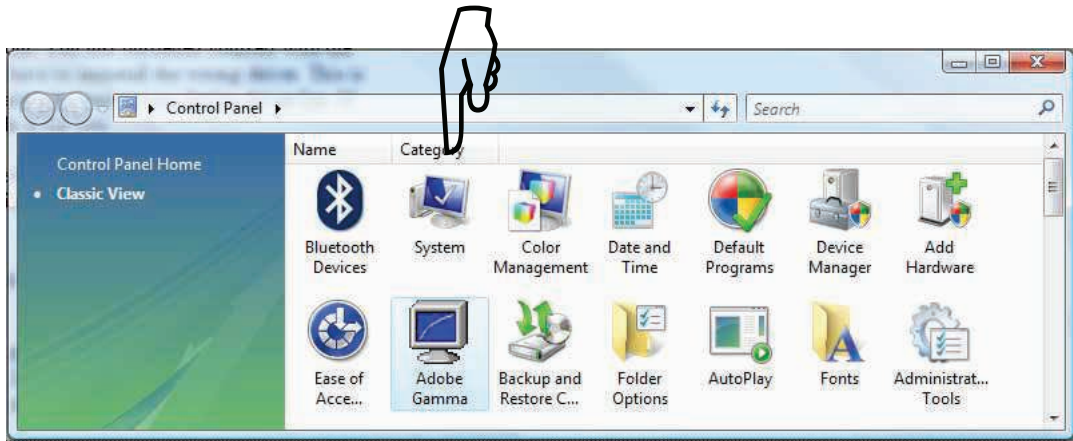


Figure 6. If your Arduino IDE suddenly runs slow on click to the Tools menu, chances are your system is choked by too many COM ports assigned to unused Bluetooth devices. Check your system COM usage by clicking Control Panel>System>Device Drivers.

Note 1:

The Arduino IDE can easily be distracted from too many unused COM port. If you use a lot of Bluetooth devices with your PC, chances are, your system is littered with too many COM port. This can slow down the port scanning of the Arduino IDE almost to a halt. You will notice this whenever you click Tools on the menu bar. It will take a long while before Arduino IDE responds with the drop menu.

To remedy this, uninstall all Bluetooth COM ports that are not in use.

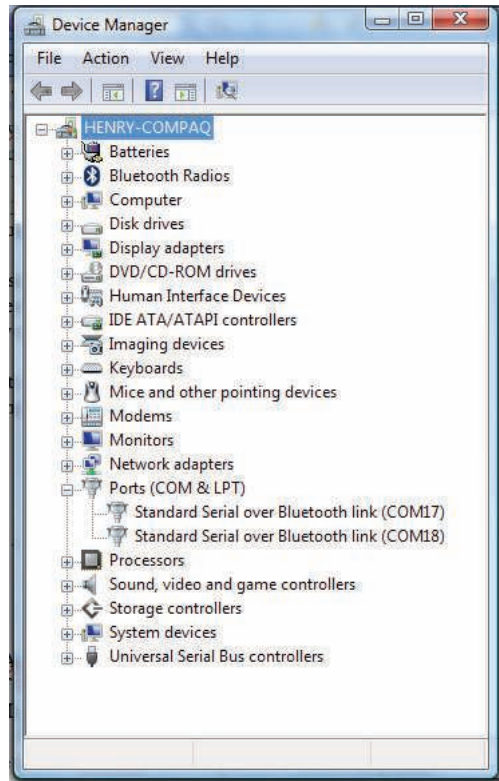


Figure 7. Click Ports to display COM ports usage. Remove all COM ports assigned to unused Bluetooth devices to speed up the Arduino.