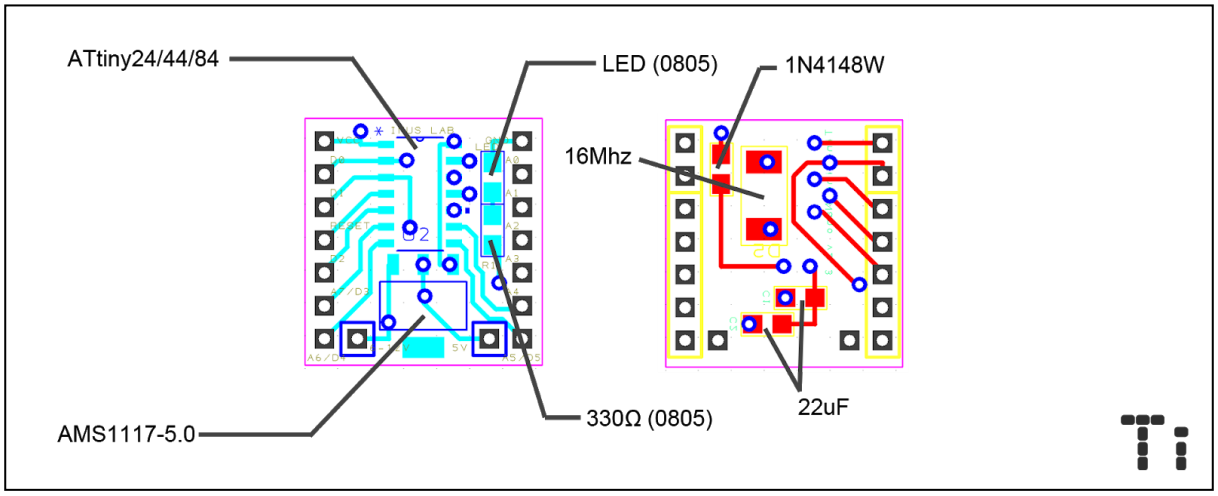
Tinduino Documentation

For Tinduino Build v2.0 or above
Support Tinduino ver 3.2+

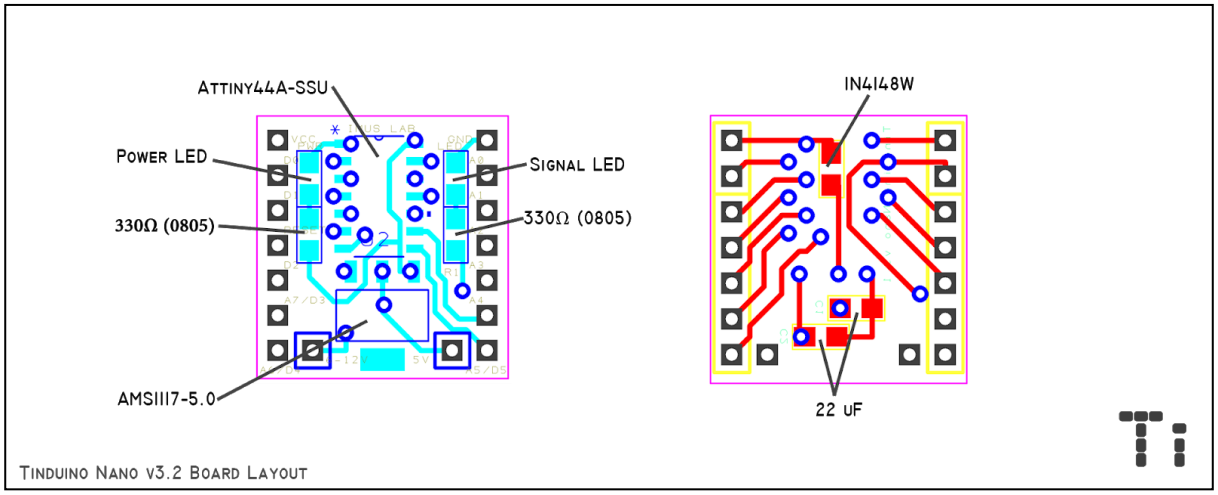
Hardware

The Tinduino is built in compatible with all ATtiny x4 series. Include but not limited to ATtiny24, ATtiny44 and ATtiny84. The hardware component for Tinduino has the following specification.

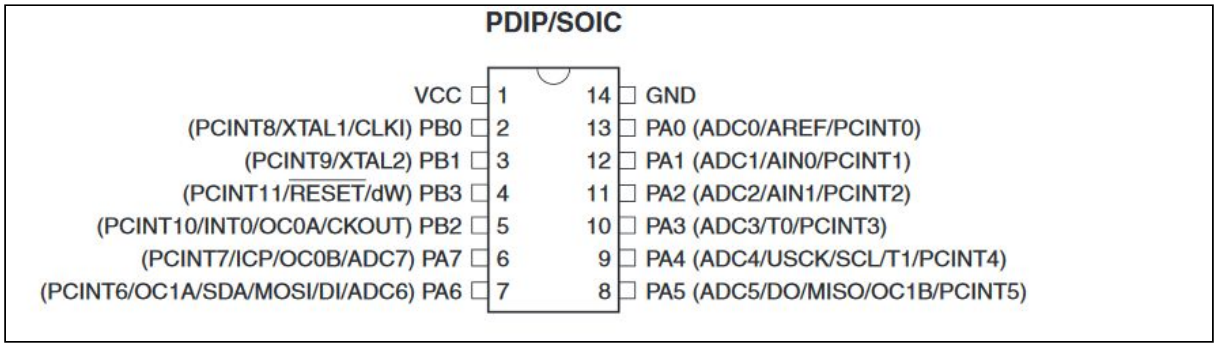
For Tinduino v2.0 to 2.3, the board layout is as follow.



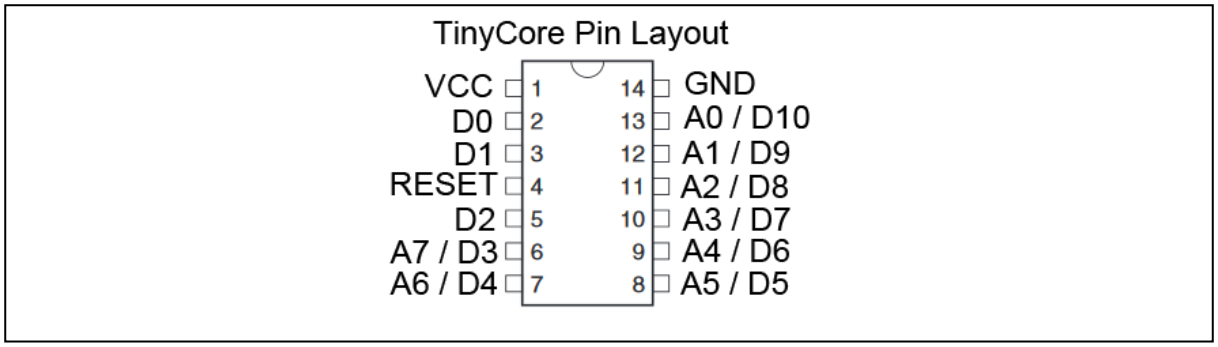
For Tinduino v3.0 to 3.2, the board layout is as follow.



Pin configuration (Provided by Atmel)

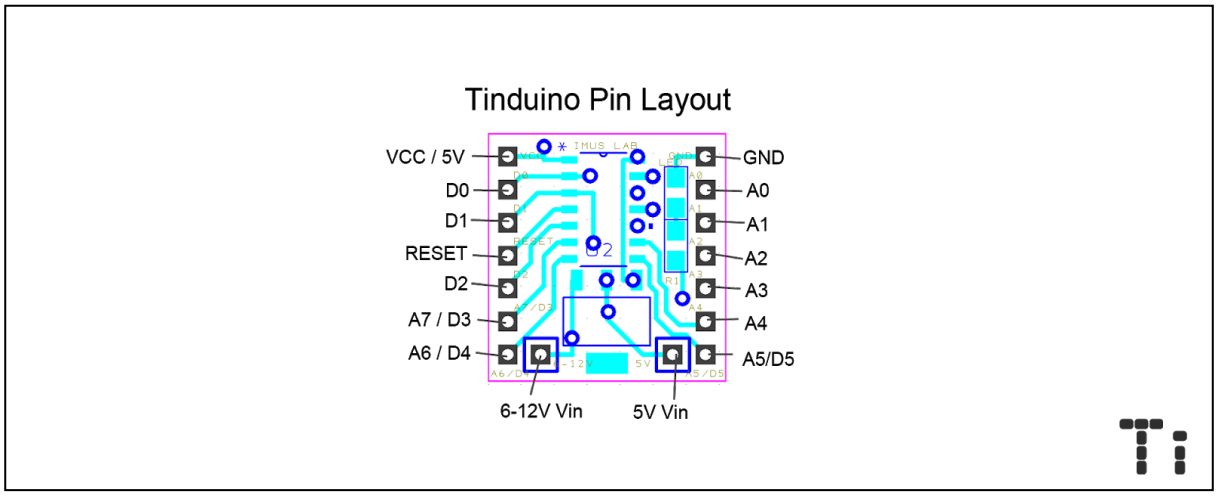


ATtiny Pin Config (Modified base on TinyCore)

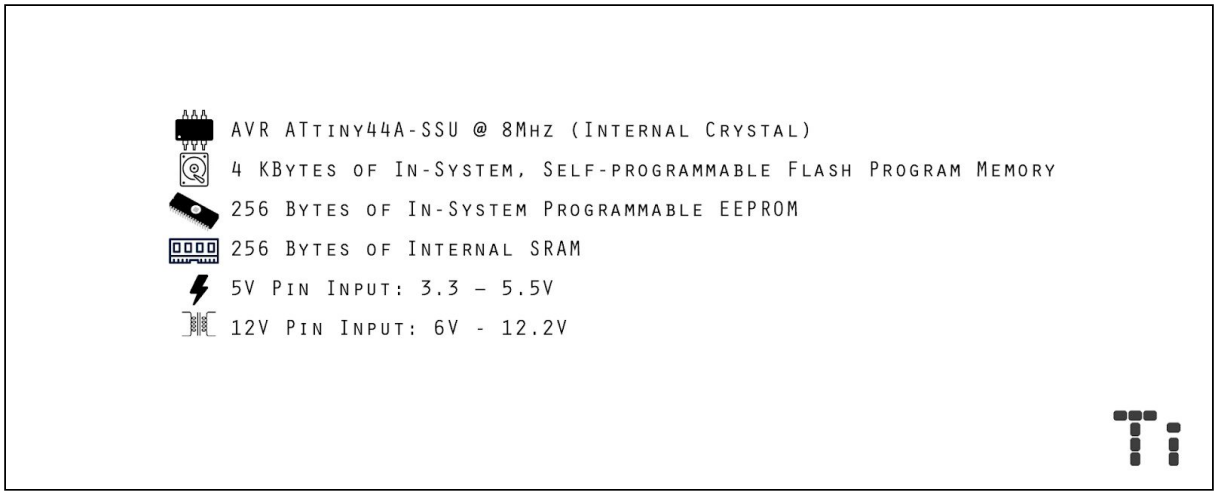


Tinduino Pin Layout

Tinduino firmware is modified from TinyCore. The pin layout of the TInduino Nano is similar to the TinyCore with the following pin configuration.



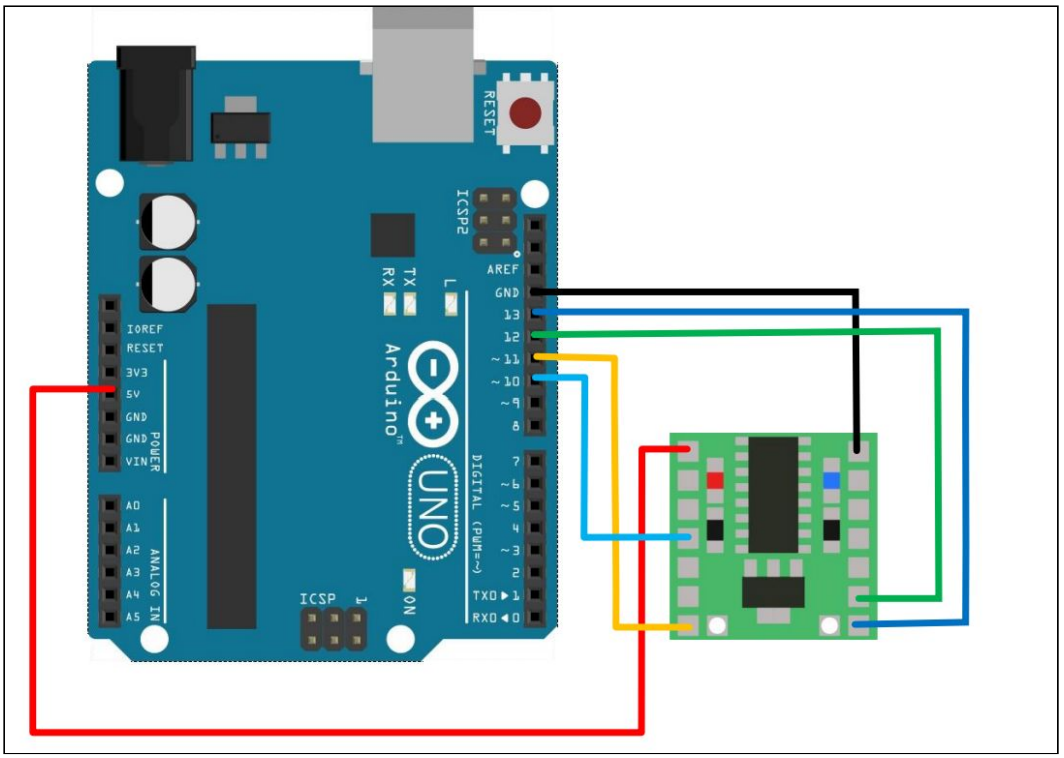
Specification



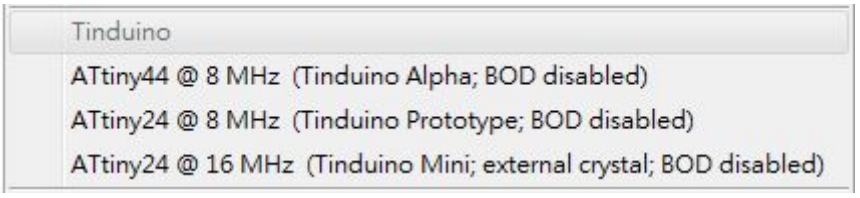
Manual Programming

Programing via Arduino ISP

1. Upload Arduino ISP to Arduino Uno or compatible Arduino
2. Connect the Tinduino to Arduino following the diagram below.



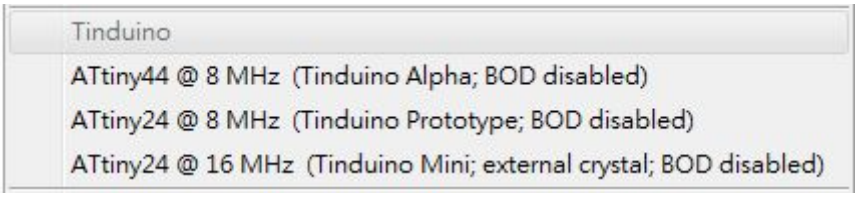
3. Select “Tinduino” from boards and select the suitable version of Tinduino you are using.



4. Upload the sketch to the Arduino that connected to Tinduino with Arduino as ISP.

Programming via AVR Programmer (USBASP)

1. Select “Tinduino” from boards and select the suitable version of Tinduino you are using and connect your Tinduino to USBASP.

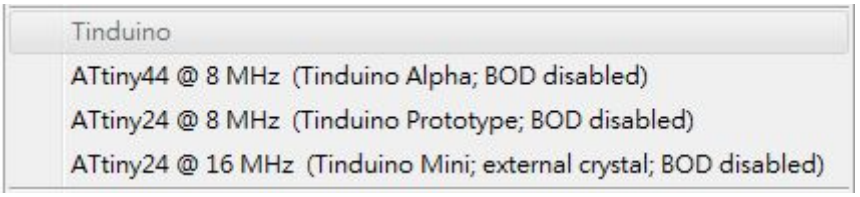


2. Compile the sketch and browse to the tmp folder which your Arduino IDE store your current sketch.
3. Copy the file with the extension “hex” to a folder which you store the “ino” sketch file as backup. (This step is not mandatory)
4. Upload the sketch to your Tinduino using USPASP option in Arduino IDE

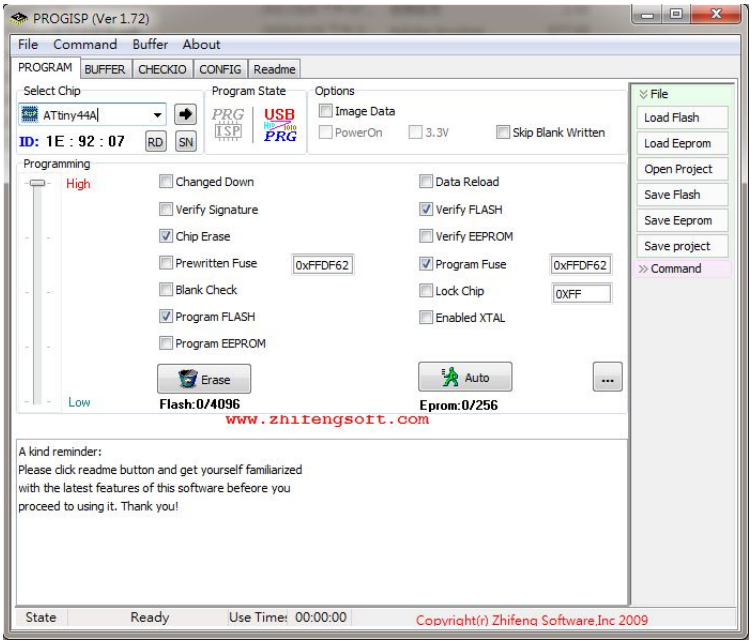
*For drivers of USBASP, please visit http://www.fischl.de/usbasp/

Programming via AVR Programmer (USBISP)

1. Select “Tinduino” from boards and select the suitable version of Tinduino you are using and connect your Tinduino to the USBISP.



2. Compile the sketch and browse to the tmp folder which your Arduino IDE store your current sketch.
3. Copy the file with the extension “hex” to a folder which you store the “ino” sketch file.
4. Open PROGISP.exe, select the chips number on your Tinduino. (e.g. ATtin44A)

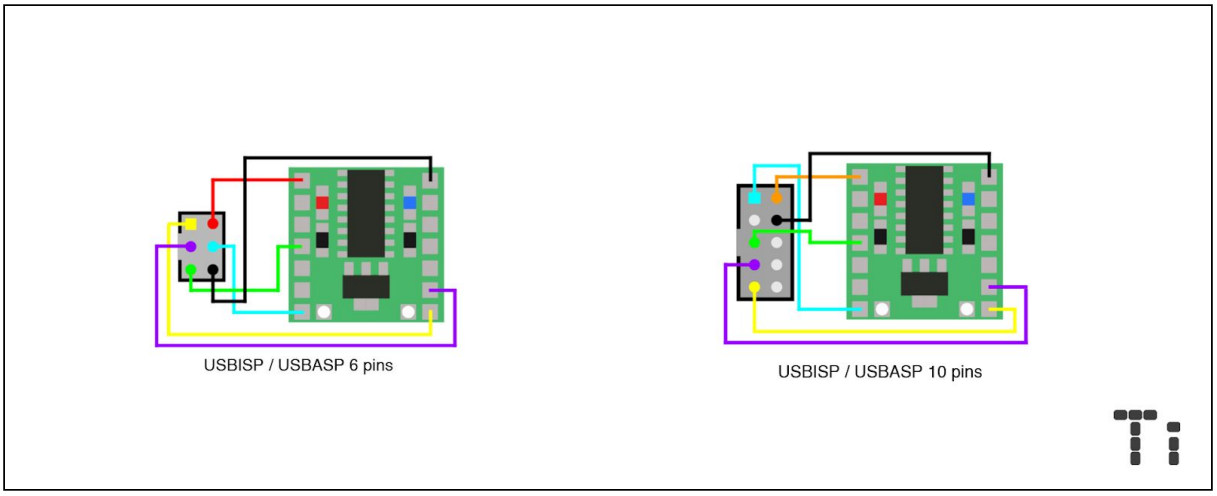


5. Click “Load Flash” from the right bar and select your “.hex” file.
6. Click “AUTO” and wait for the write process to be finished.

*PROGISP can be downloaded at http://www.electrodragon.com/w/ProgISP

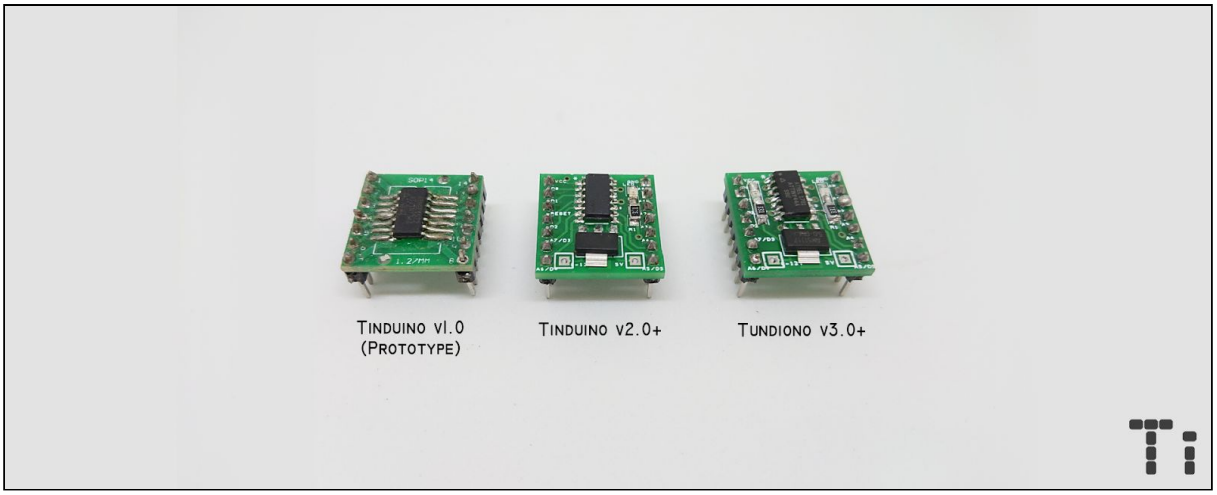
Wiring for Programming

The following wiring diagram is written for both USBISP and USBASP.



Version Identification

You can identify the Tinduino your own by observing the version label printed on the PCB on the bottom side of the Tinduinos. You can also observe the main version number of your Tinduino by the board layouts listed in the graph below.

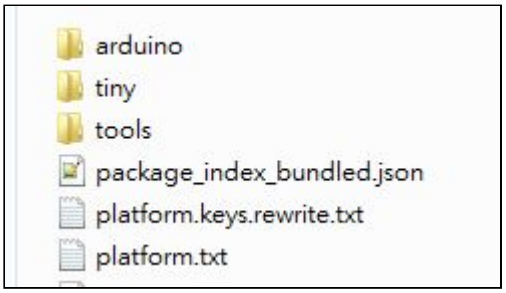


Online Compiler

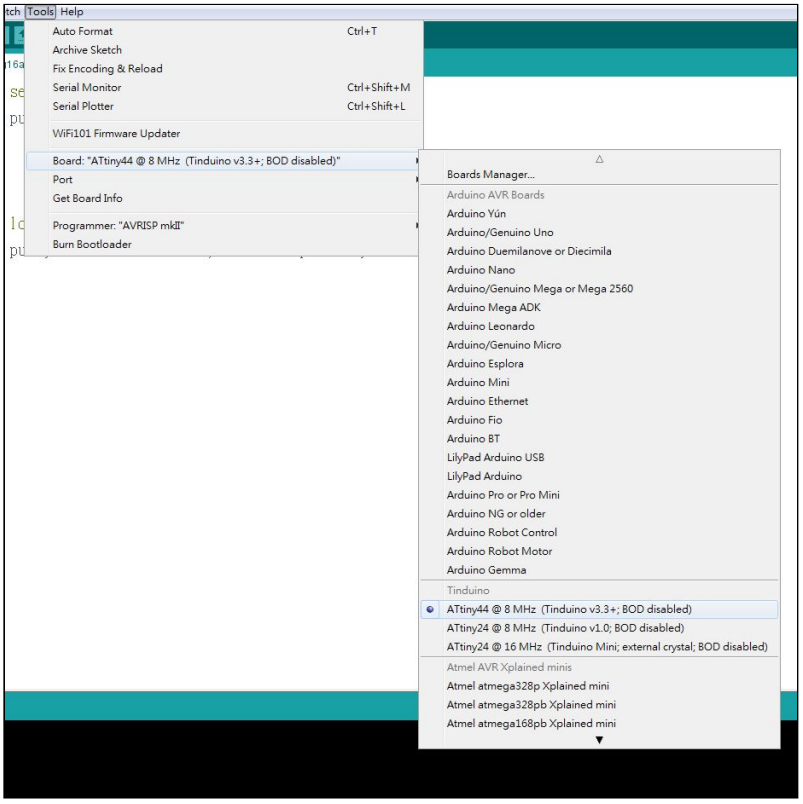
*The online compiler is still in experimental build. Please use with your own risk.
http://tinduino.com/compiler or https://tid.rubm.me/compiler/

Setting up Arduino IDE for Tinduino

1. Download the board info file from http://download.hkwtc.org/tiny.zip or http://imuslab.com/tind/download/tiny.zip .ATtinyCore from https://github.com/SpenceKonde/ATTinyCore can also be used in Tinduino.
2. Unzip it into “hardware” under your Arduino IDE’s root folder.
(Usually “C:\Program Files (x86)\Arduino\hardware” if you are running x64 version of windows OR “C:\Program Files\Arduino\hardware” if you are running x86 version of windows)
3. After unzipping, you will see at least 3 folders and some other files generated by the Arduino IDE as follow.



4. Launch the Arduino IDE, look for the “Tinduino” section in the Tools/board/ list.



5. If the “Tinduino” section exists, it means that you have installed Tinduino into Arduino IDE successfully.

Advance Board Information Editing

During the process of installing Tinduino into the Arduino IDE, if your Arduino IDE already exists a folder named “tiny”, please do not overwrite the folder or files and follow the instruction below to prevent any bugs created by incompatibility of alternative Arduino compatible boards.

> The reason for the existence of “tiny” folder is due to the fact that Tinduino driver and bootloader was modified from the open source “TinyCore” bootloader. Hence, overwriting the existing “tiny” folder will lead to problems if the original TinyCore in your PC have different version of bootloaders.

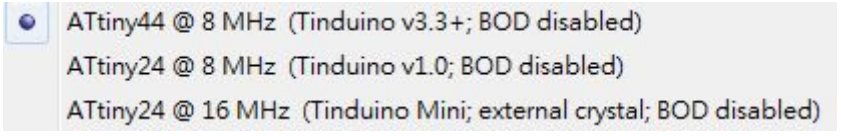
Method 1

1. Unzip the “tiny” folder somewhere else and rename it as “Tinduino”
2. Move the “Tinduino” folder into the “hardware” directory under the root of the Arduino IDE.
3. Check if the installation was successfully.

Method 2

1. Open “boards.txt” under the “tiny” folder downloaded from the Tinduino site.
2. Copy ALL lines in the Tinduino’s “boards.txt” that contains no “#”
3. Open “boards.txt” from your original “tiny” folder.
4. Paste all the text from your clipboard into the bottom part of your original “boards.txt”
5. Save and Exit.
6. Check if the installation was successful

For this installation method, the section name will not show as “Tinduino”. However, Tinduino boards setting are still being added into the board list as shown in the picture below.



Or select Tinduino v3.3 or above if you have a newer versions of the ‘tiny folder’ setup files.