

AZ-Delivery

Welcome!

Thank you for purchasing our *AZ-Delivery Atmega328 Pro Mini 5V*. On the following pages, we will introduce you to how to use and set-up this handy device.

Have fun!

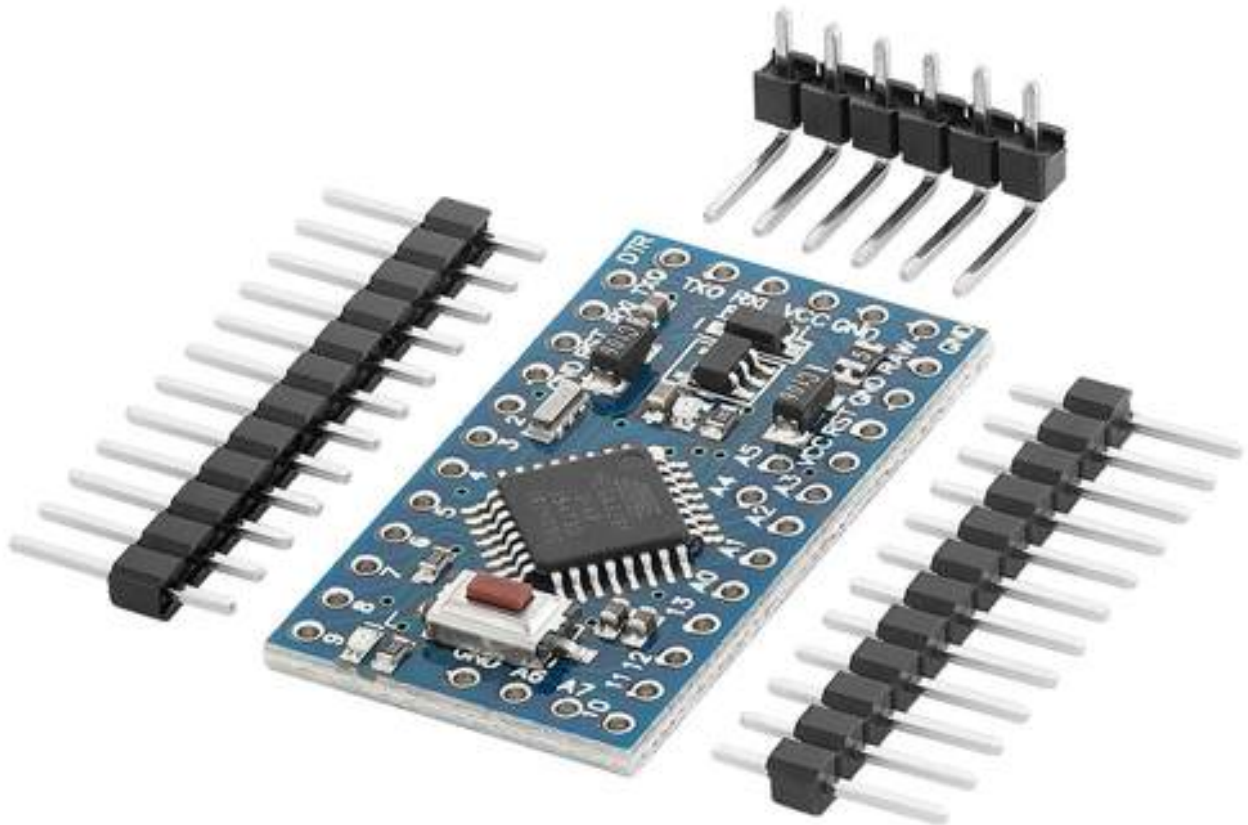


Table of Contents

Introduction.....	3
Specifications.....	4
ATmega328.....	4
Pinout.....	5
I/O pins.....	6
Serial communication pins.....	6
Analog pins.....	6
I2C pins.....	6
SPI pins.....	7
On-board LEDs.....	7
How to set-up Arduino IDE.....	8
Programming the Atmega328 Pro Mini.....	12
Connecting the Atmega328 Pro Mini with Atmega328p.....	15
Connecting the Atmega328 Pro Mini with FTDI adapter.....	16
Sketch examples.....	17
Blinking LED.....	17
PWM - Pulse Width Modulation.....	18



Introduction

The Atmega328 Pro Mini is a small microcontroller board that is based on ATmega328P Microcontroller. It features similar connectivity and functionality found in other Atmega328p boards. The Atmega328 Pro Mini small size and flexible nature makes it one of the most preferable boards, especially by advanced users.

It is a smaller version of Atmega328p. It uses the same chip and it is much smaller in size. Atmega328 Pro Mini has the same capabilities as the Atmega328p except it has no on-board programmer chip.

The Atmega328 Pro Mini size makes it very usable when microcontroller that requires very little space, which makes Atmega328 Pro Mini perfect candidate for compact size projects. Programming the Atmega328 Pro Mini requires an external programmer device and several devices can be used for this purpose.

It can be programmed with a FTDI adapter or USB to TTL converter. In case that one does not have the FTDI adapter or USB to TTL converter, programming can be simply done with an Atmega328p.

With a few wires properly connected the Atmega328 Pro Mini can be programmed in minutes. The Arduino IDE is perfect solution which makes programming the Atmega328 Pro Mini very easy.

Specifications

Microcontroller	ATmega328
Onboard voltage regulator	5V
Input voltage limit	6V-16V (RAW pin)
Max. current from single I/O	40mA
Max. current drawn from chip	200mA
Recommended current per pin	20mA
CPU frequency	16MHz
Digital I/O pins	14
Analog input pins	8
Flash memory	32KBytes
RAM	2KB
EEPROM	1KB
Clock speed	16MHz
Communication interfaces	SPI, I2C, UART
Dimensions	33x18x2mm (1.3x0.7x0.1in)

There are 2 versions of Atmega328 Pro Mini on the market. Main difference is the on-board voltage regulator.

This version of Atmega328 Pro Mini has an on-board 5V voltage regulator that regulates the voltage from the *RAW* pin. Maximum unregulated voltage that can be applied through this pin can be 16VDC.

Az-Delivery

I/O pins

Serial communication pins

Serial pins RXD and TXD are used to transmit and receive serial data. The RX receives the data and TX is used to transmit data.

There is one UART interface (Universal Asynchronous Receiver Transmitter) where Serial 0 contains RX(0) and TX(1).

Analog pins

There are 8 analog pins incorporated on the board and labeled from A0 to A7. Each analog pin comes with 10bit resolution.

I2C pins

Two pins A4 and A5 support I2C communication where A4 represents SDA (Serial Data Line), mainly used for holding the data and A5 represents SCL (Serial Clock Line), mainly used for providing data synchronization between the devices.

Az-Delivery

SPI pins

The SPI stands for Serial Peripheral Interface used for the transmission of data between the controller and other peripheral components. Four pins 12(MISO), 11(MOSI), 13(SCK), 10(SS) are used for SPI communication.

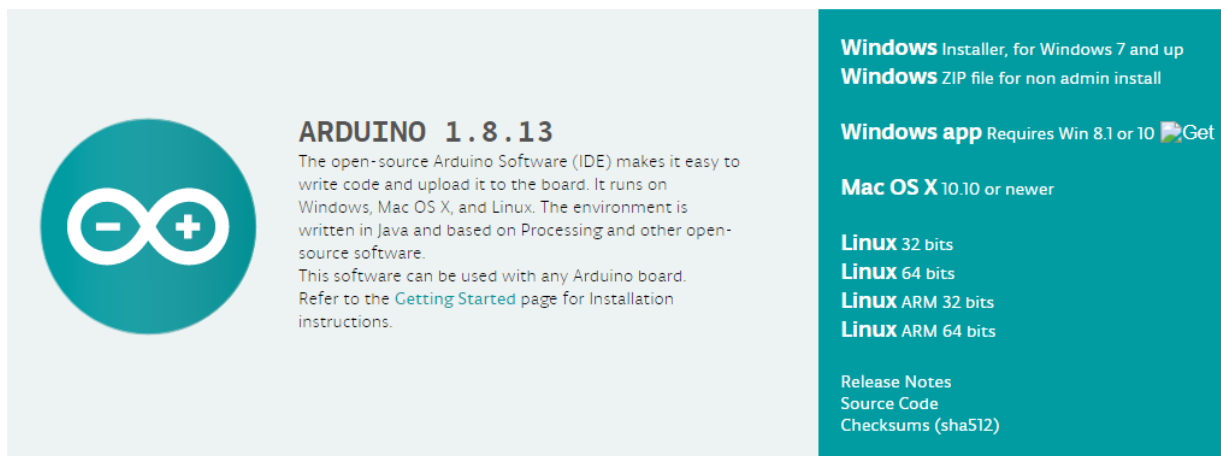
On-board LEDs

This board comes with two built-in LEDs. One LED is connected to digital pin 13. HIGH value at this pin will turn the LED on and LOW value will turn it off. Simple *Blink* example from the Arduino IDE can be used for testing the Atmega328 Pro Mini for the first time.

How to set-up Arduino IDE

If the Arduino IDE is not installed, follow the [link](#) and download the installation file for the operating system of choice. The Arduino IDE version used for this eBook is **1.8.13**.

Download the Arduino IDE



ARDUINO 1.8.13

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

Windows Installer, for Windows 7 and up
Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10 [Get](#)

Mac OS X 10.10 or newer

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

[Release Notes](#)
[Source Code](#)
[Checksums \(sha512\)](#)

For *windows* users, double click on the downloaded .exe file and follow the instructions in the installation window.

Az-Delivery

For *Linux* users, download a file with the extension `.tar.xz`, which has to be extracted. When it is extracted, go to the extracted directory and open the terminal in that directory. Two `.sh` scripts have to be executed, the first called `arduino-linux-setup.sh` and the second called `install.sh`.

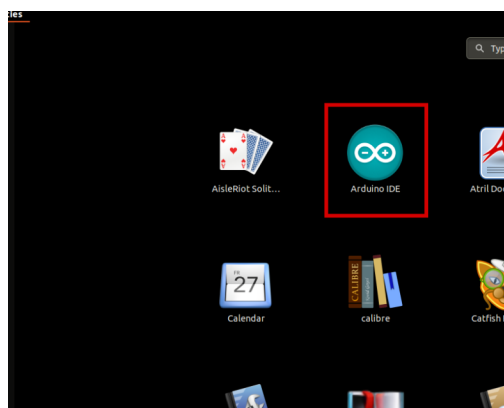
To run the first script in the terminal, open the terminal in the extracted directory and run the following command:

```
sh arduino-linux-setup.sh user_name
```

user_name - is the name of a superuser in the Linux operating system. A password for the superuser has to be entered when the command is started. Wait for a few minutes for the script to complete everything.

The second script, called `install.sh`, has to be used after the installation of the first script. Run the following command in the terminal (extracted directory): **sh install.sh**

After the installation of these scripts, go to the *All Apps*, where the *Arduino IDE* is installed.



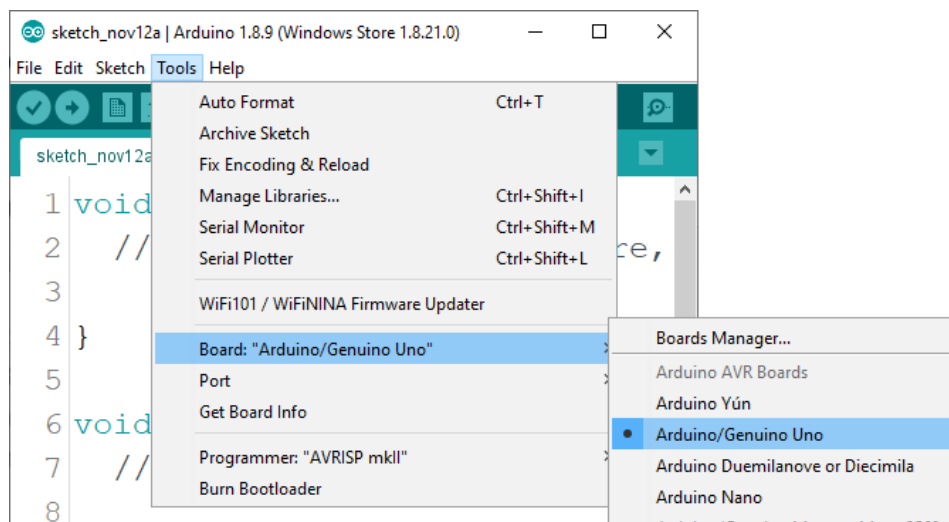
Az-Delivery

Almost all operating systems come with a text editor preinstalled (for example, *Windows* comes with *Notepad*, *Linux Ubuntu* comes with *Gedit*, *Linux Raspbian* comes with *Leafpad*, etc.). All of these text editors are perfectly fine for the purpose of the eBook.

Next thing is to check, if your PC can detect an Atmega328p board. Open freshly installed Arduino IDE, and go to:

Tools > Board > {your board name here}

{your board name here} should be the *Arduino/Genuino Uno*, as it can be seen on the following image:



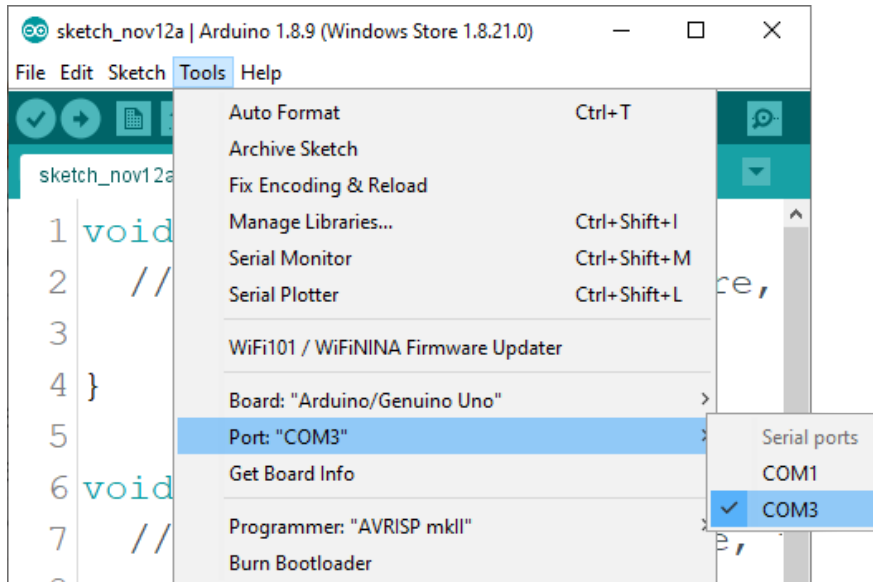
The port to which the Atmega328p board is connected has to be selected.

Go to: *Tools > Port > {port name goes here}*

and when the Atmega328p board is connected to the USB port, the port name can be seen in the drop-down menu on the previous image.

Az-Delivery

If the Arduino IDE is used on Windows, port names are as follows:



For *Linux* users, for example port name is `/dev/ttyUSBx`, where *x* represents integer number between 0 and 9.



Programming the Atmega328 Pro Mini

The Atmega328 Pro Mini can be programmed in several ways. For this solution the FTDI adapter can be used for USB to Serial communication or the USB to TTL converter. In case that neither is available, the Atmega328p can be used. Using Atmega328p for programming the Atmega328 Pro Mini is the easiest way (assuming that one already has the Atmega328p). Only requirement is to carefully remove the Atmega chip from the Atmega328p socket. This way, the on-board USB to Serial programmer is used through the TX, RX pins.

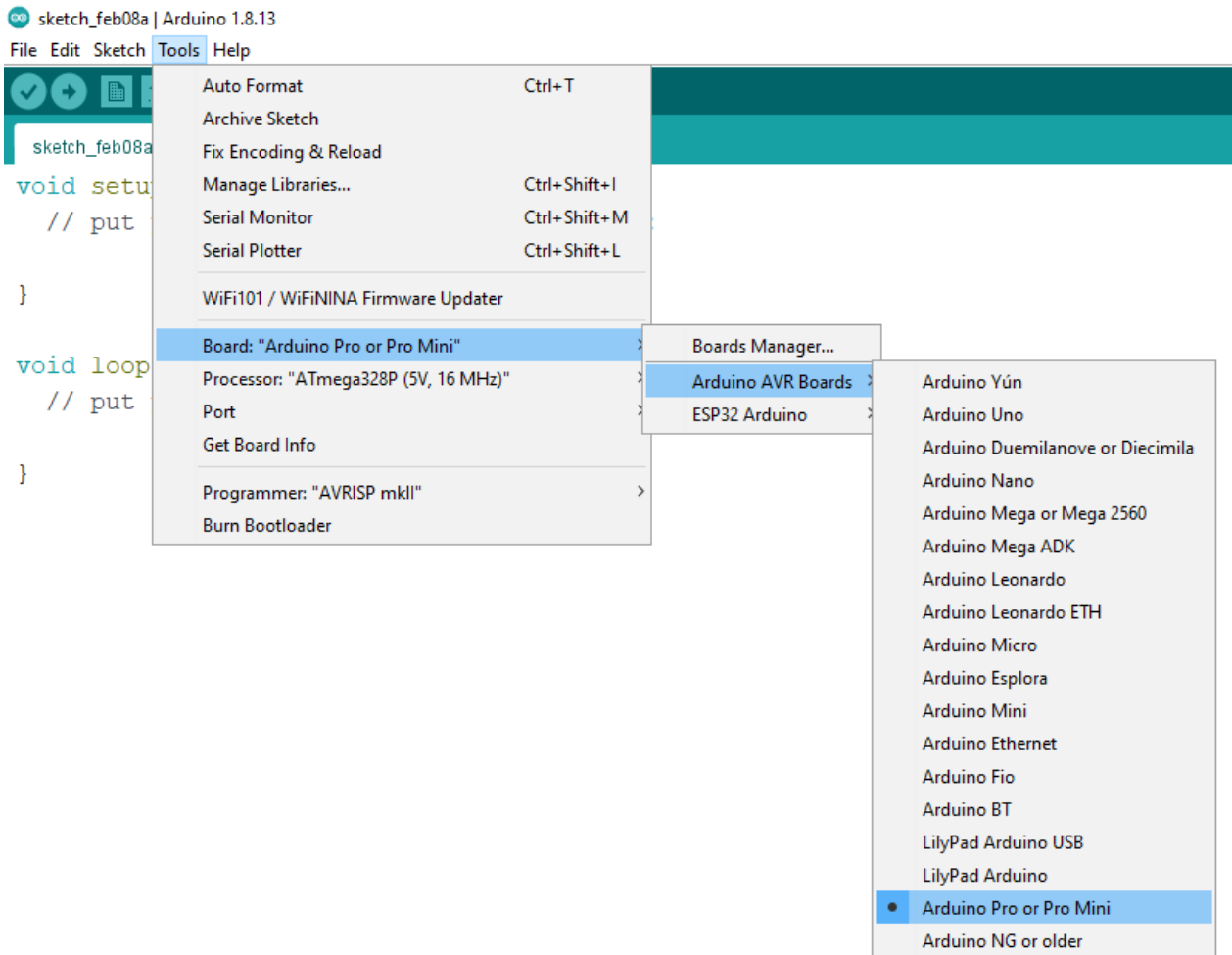
For testing purpose simple Blink sketch can be uploaded and when on-board LED starts flashing it is the signal that the upload is successful.

In some cases the reset pin has to be used and some devices such as USB to TTL converter do not have one. However the USB to TTL converter can be moded by carefully soldered wire to the pin 28 of the on-board chip. See the [datasheet](#) for CP2102 chip.

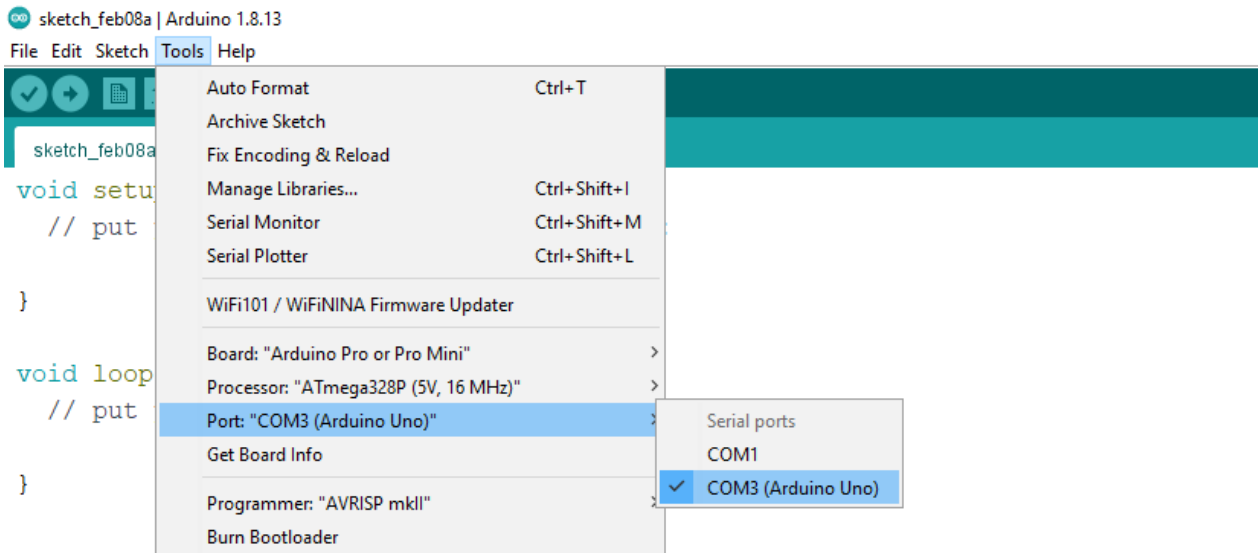
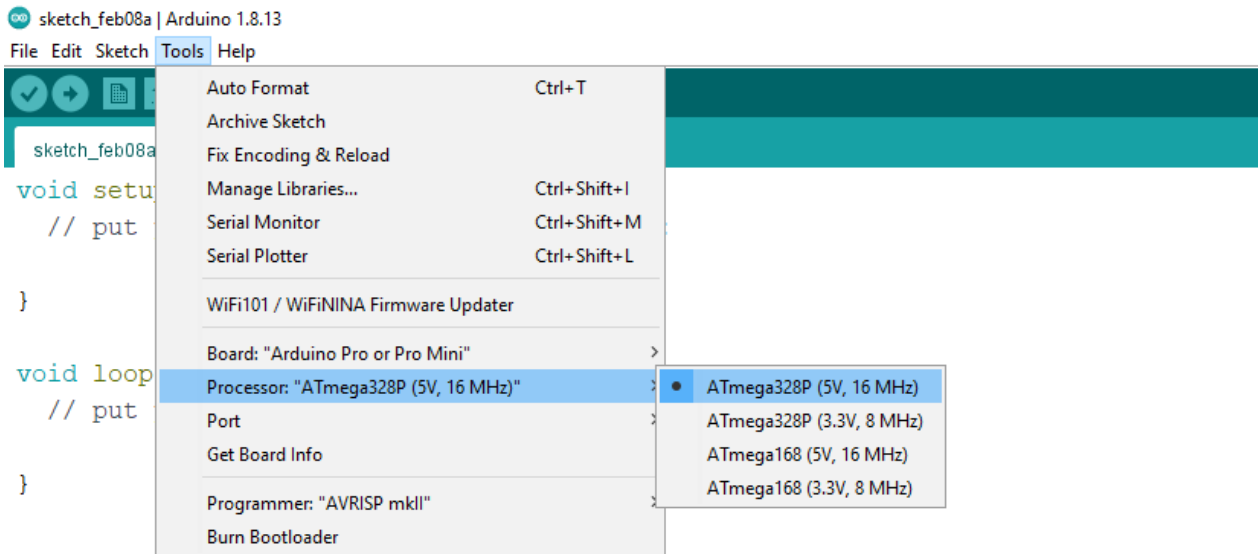
If these procedures are too much hassle, simply use the Atmega328p.

Az-Delivery

To program the Atmega328 Pro Mini, set the board from IDE menu as on the following images:

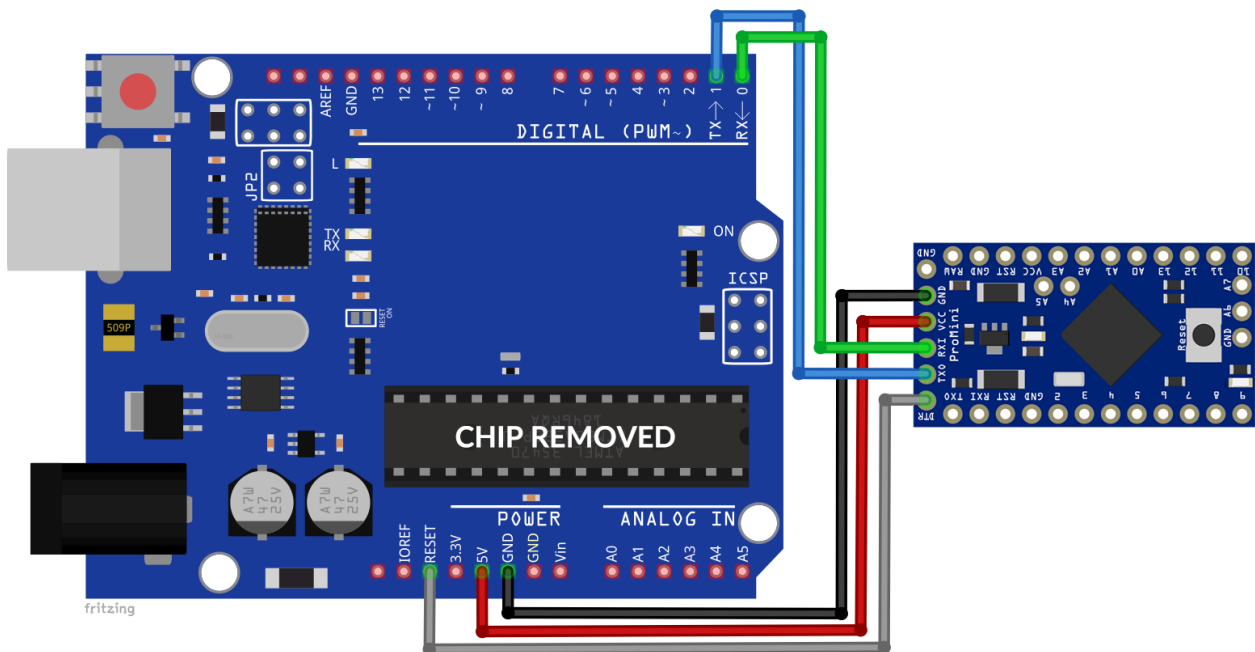


Az-Delivery



Connecting the Atmega328 Pro Mini with Atmega328p

Connect the Atmega328 Pro Mini with Atmega328p as shown on the following image:



Atmega328 Pro Mini pin	Mc pin	Wire color
GND	GND	Black wire
VCC	5V	Red wire
RXI	RX	Green wire
TXO	TX	Blue wire
DTR	RESET	Gray wire

Az-Delivery

Sketch examples

Blinking LED

```
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH);  
  delay(1000);  
  digitalWrite(LED_BUILTIN, LOW);  
  delay(1000);  
}
```

PWM - Pulse Width Modulation

```
int led = 9;
int brightness = 0;
int fadeAmount = 5;

void setup() {
  pinMode(led, OUTPUT);
}

void loop() {
  analogWrite(led, brightness);
  brightness = brightness + fadeAmount;

  if (brightness <= 0 || brightness >= 255) {
    fadeAmount = -fadeAmount;
  }
  delay(30);
}
```

To test the *Fade* sketch, connect the LED between GND and digital pin 9 in series with a resistor.

AZ-Delivery

Now it is the time to learn and make your own projects. You can do that with the help of many example scripts and other tutorials, which can be found on the Internet.

If you are looking for the high quality microelectronics and accessories, AZ-Delivery Vertriebs GmbH is the right company to get them from. You will be provided with numerous application examples, full installation guides, eBooks, libraries and assistance from our technical experts.

<https://az-delivery.de>

Have Fun!

Impressum

<https://az-delivery.de/pages/about-us>