

V1 Arduino firmware load.

There are two versions of the firmware for the Arduino based shutter tester.

Both are identical, only differing in which type of receiver sensor is used.

Original sensors gave an output when the Laser is seen. Newer sensors reversed this, giving an output when the Laser is not seen.

There is no way to tell which type of sensor you have. It is suggested to use the Arduino_shuttertimer_x_x_x_new.hex file.

The tester incorporates a Laser alignment utility, which will show which type of sensors you have. If your sensors are working in reverse, load the ori.hex file.

If not already done so, download the code from the github page.

click on the green <>Code button, which will allow you to download all of the files as a zip file.

Un-zip the downloaded file. You will find two versions of Arduino code. 'ori' for the original laser receivers and 'new' for the newer type.

(note:- there will also be an ESP32 version of the code, do not try loading this into an Arduino!).

Flashing firmware onto the Arduino Board.

A program called AVRDUDESS is required. This can be downloaded from

<https://github.com/ZakKemble/AVRDUDESS/releases/download/v2.14/AVRDUDESS-2.14-setup.exe>

To view the web page (for those who want more details

[AVRDUDESS – A GUI for AVRDUDE | Zak's Electronics Blog \(zakkemble.net\)](#)

Watch this video, from 2.30 to 5.15

<https://youtu.be/Wcaql0jtlUg>

It explains how to load the .hex file onto your Arduino. Watch the video first, then read the below before trying to load the .hex file to your Arduino. *Ignore the first & last part of the video, it is not relevant. Watch between 2.30 and 5.15*

At 3.31, connect your Arduino to your computer using an appropriate USB cable. The drop-down menu in AVRDUDESS should find the correct com port, if not, go to Device Manager (press Windows Key + x then select Device Manager) on your computer to find which com port has been assigned to the Arduino board.

Note:- If the correct driver is not on your computer, you will need to download and install it. Most Chinese Nano clones use the CH340 driver

At 3.34 this is where you browse to your downloaded and un-zipped code download and select one of the .hex files

A4 4.18 'Arduino Uno (Atmega328P)' is selected. ***As Saravanan says this is very important ***.***

You will also notice, when selecting this, the com port changes to 1 and the file path disappears. BE SURE to select the correct COM port and file path again.

*** To complicate things, there are two different bootloaders which could be in your Nano.

Most Chinese Nano boards have the old bootloader.

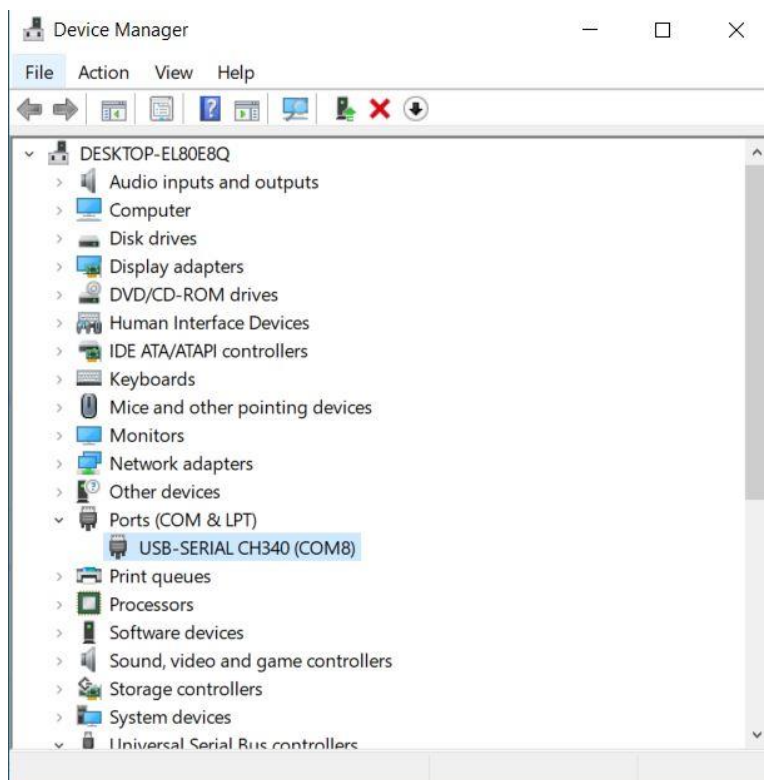
For the old bootloader at 4.18, you will need to select 'Arduino Nano (ATmega328P)' in the Preset box.

For the new bootloader, select 'Arduino Uno (ATmega329P)' as shown in the video.

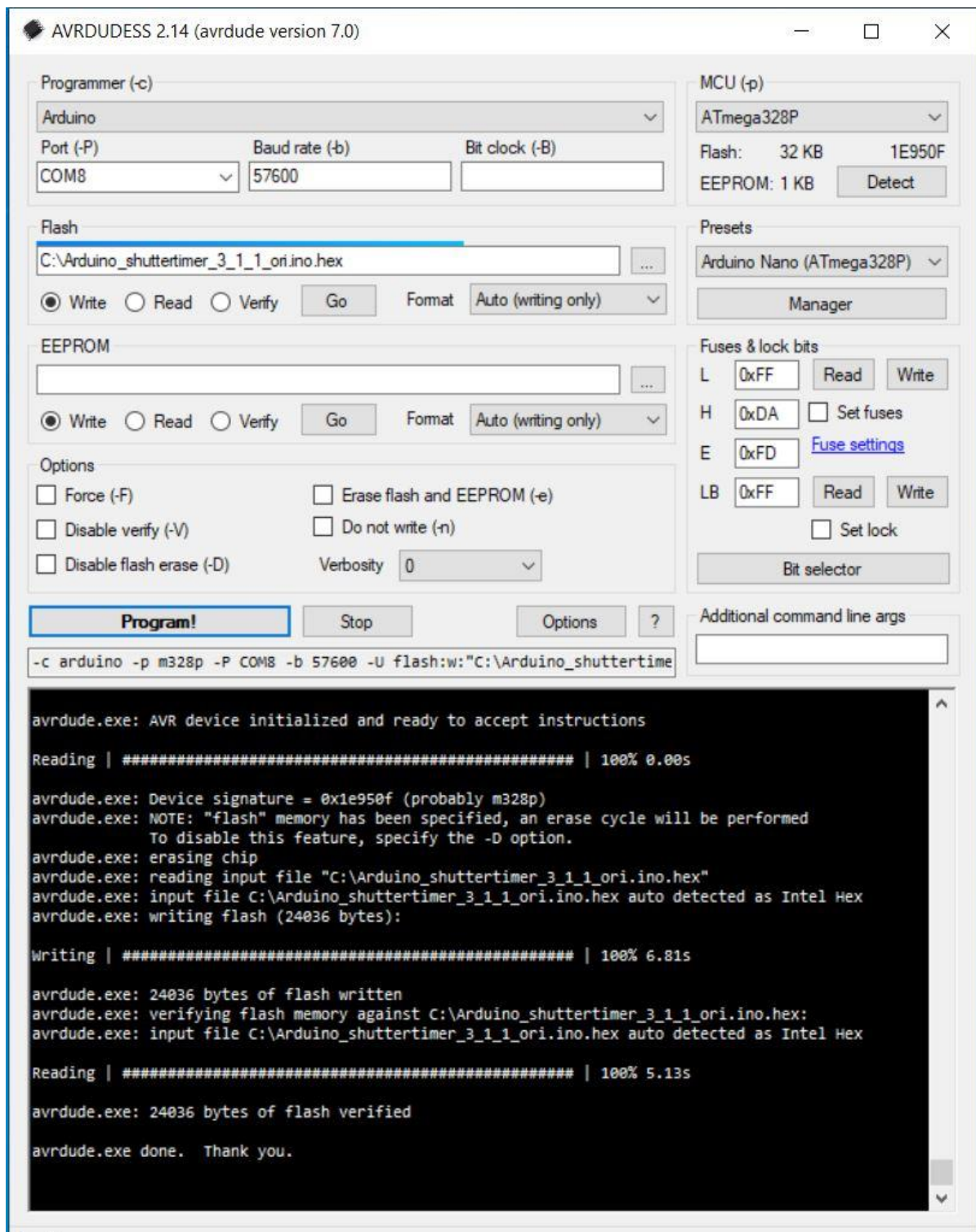
It is suggested to select 'Arduino Nano (ATmega328P)' in the Preset box, as the first option. If this does not work, then try 'Arduino Uno (ATmega328P).'

Below are four screenshots,

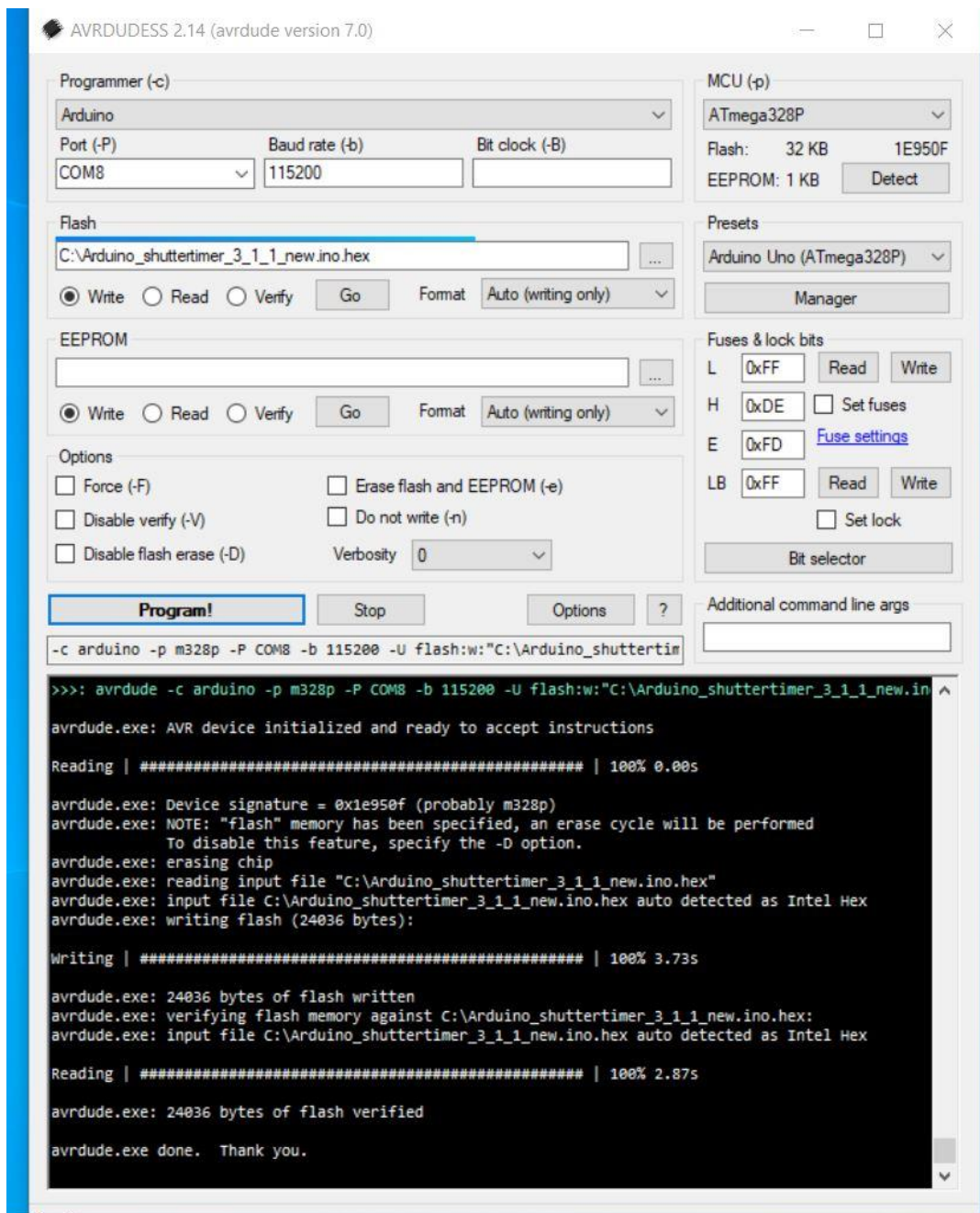
- 1) showing the com port in Device Manger (your com port number will be different),
- 2) Loading software as Nano (old bootloader)
- 3) Loading software as Uno (new bootloader)
- 4) Loading with wrong bootloader selected, showing errors.



Device Manger



Loading software as Nano (old bootloader)



Loading software as Uno (new bootloader)

Programmer (-c)

Arduino

Port (-P)

COM8

Baud rate (-b)

115200

Bit clock (-B)

Flash

C:\Arduino_shuttertimer_3_1_1_ori.ino.hex

...

☒ Write

☐ Read

☐ Verify

Go

Format

Auto (writing only)

EEPROM

...

☒ Write

☐ Read

☐ Verify

Go

Format

Auto (writing only)

Options

☐ Force (-F)

☐ Erase flash and EEPROM (-e)

☐ Disable verify (-V)

☐ Do not write (-n)

☐ Disable flash erase (-D)

Verbosity

0

Program!

Stop

Options

?

-c arduino -p m328p -P COM8 -b 115200 -U flash:w:"C:\Arduino_shuttertim

MCU (p)

ATmega328P

Flash: 32 KB

1E950F

EEPROM: 1 KB

Detect

Presets

Arduino Uno (ATmega328P)

Manager

Fuses & lock bits

L

0xFF

Read

Write

H

0xDE

☐ Set fuses

E

0xFD

[Fuse settings](#)

LB

0xFF

Read

Write

☐ Set lock

Bit selector

Additional command line args

```

>>>: avrdude -c arduino -p m328p -P COM8 -b 115200 -U flash:w:"C:\Arduino_shuttertimer_3_1_1_ori.ino.hex"
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 1 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 2 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 3 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 4 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 5 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 6 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 7 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 8 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 9 of 10: not in sync: resp=0x88
avrdude.exe: stk500_recv(): programmer is not responding
avrdude.exe: stk500_getsync() attempt 10 of 10: not in sync: resp=0x88
avrdude.exe: opening programmer "arduino" on port "COM8" failed

avrdude.exe done. Thank you.

```

Loading with wrong bootloader selected, showing errors.