## Programming Arduino Nano Every from Microchip AVR Studio *without* a separate programmer

(Version 1.0, 10/08/2022, CMN)

### Introduction

You can program the ATmega4809 microcontroller used in the Arduino Nano Every in the same way as you do in the Arduino IDE, but it needs to be set up first. In this tutorial, I assume that you are using an Arduino Nano Every with a USB Micro-B cable. Unlike the ATmega328P in an Arduino UNO, The Nano Every ATMega4809 does not use a Bootloader. Instead, it uses its ‘Universal Programming and Debug Interface,’ (UPDI), which is controlled by other microcontroller on the Nano Every, the ATSAMD11.

**Note** that although we do not use it explicitly it is a good idea to install Arduino software (<https://www.arduino.cc/en/Main/Software>) to make sure that the latest drivers for the various Arduino boards are installed on your PC.

### Procedure (what to do)

This sequence uses a program called ***avrdude.exe*** to do the programming and you have to tell Microchip Studio where to find it and give the program parameters to make it work. In this tutorial we use Avrdude Version 7.0 because it has support for the UPDI used to program the ATmega4809, but later versions should also work well.

### Download the avrdude program

You can find the latest version of avrdude at <https://github.com/mariusgreuel/avrdude/releases>. Download the zip file, [avrdude-v7.0-windows-windows-x64.zip](https://github.com/mariusgreuel/avrdude/releases/download/v7.0-windows/avrdude-v7.0-windows-windows-x64.zip), to a directory or folder on your PC. I downloaded them to C:\avrdude\_7\_0 on my PC. That is why you see this path in the examples below.

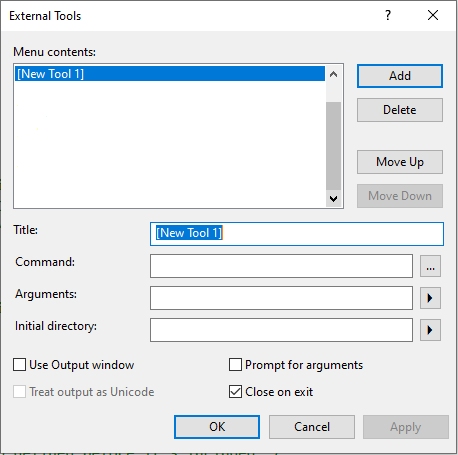
Unzip the zip file in the avrdude\_7\_0 directory or folder.

Next, from the github site under /src, download the batch file called programFlash.bat to your C:\avrdude\_7\_0 directory or folder.

If you decide to use a different folder, it is a good idea to use a folder that you can easily find and that has a short filename, preferably with no spaces in the path name.

### Setting up Microchip Studio for Arduino Nano Every programming using avrdude

1. From the Microchip Studio menu select Tools -> External Tools
2. You will see something like see the following screen:



1. Give the Tool a title (I chose Program) then enter text exactly as shown below in the remaining fields. The text to enter is as follows:

**Command:**

c:\avrdude\_7\_0\programFlash.bat

Here you are providing the full path to the programFlash.bat batch file, which invokes the avrdude.exe program. Note that C:\avrdude\_7\_0 is the folder where I downloaded and unzipped the avrdude program and configuration files. If you used a different folder, then use that folder instead of avrdude\_7\_0.

If your path name includes spaces (e.g. in Program Files, then the Command sequence needs to be in double quotes. Otherwise, you do not need them).

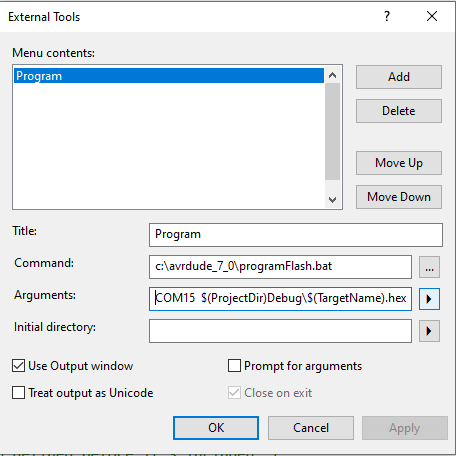
**Arguments:**

The arguments are what avdude uses to control the programming, and they are provided to avrdude.exe by the programFlash.bat batch file.

For an Arduino **Nano Every** you can copy and paste the following line, but you may need to modify it for the COM port in your PC setup (I used COM15, but this will almost certainly be different in your PC):

COM15 $(ProjectDir)Debug\$(TargetName).hex

Make sure that the ‘Use output Window’ is ticked. You will see something like the following:



We’re almost ready to go – we just need to correct the COM port. Almost every time you connect a new Arduino to a PC, a new COM port is assigned to that Arduino.

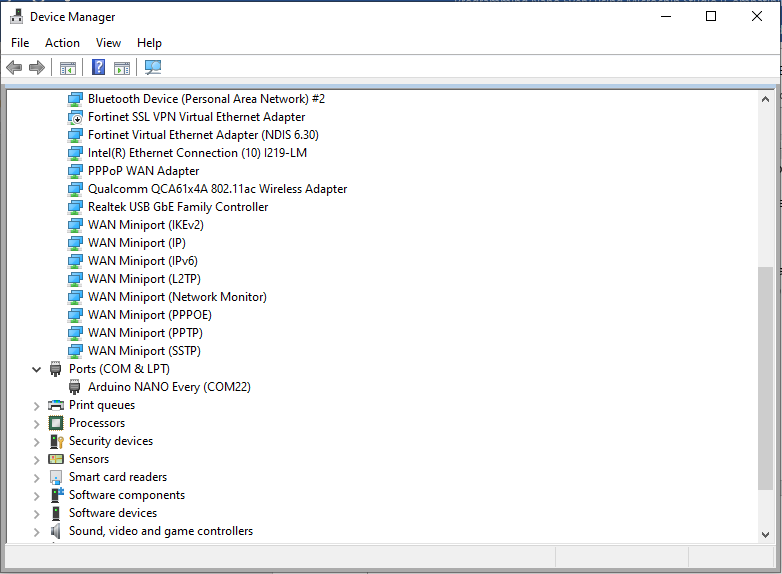
### COM Port.

Use the PC’s Device Manager to check what COM port is being used by your Arduino board.

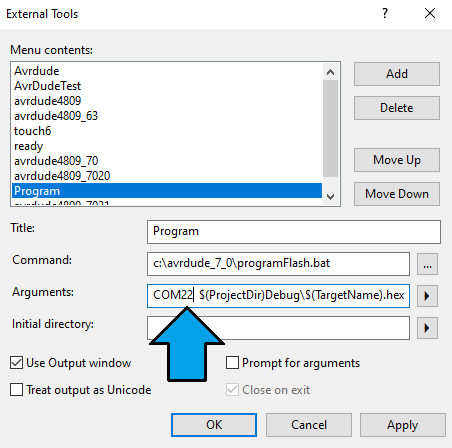
From the Windows search box, type ‘device manager’, to open the Device Manager.

In Device Manager, click on Ports (COM & LPT):

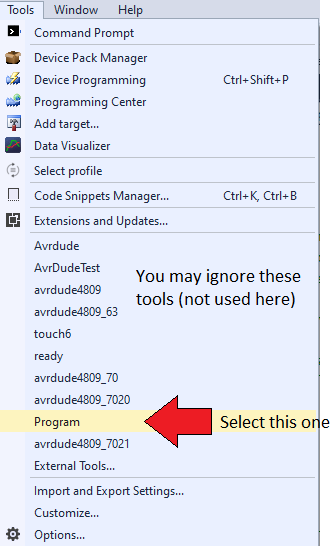
In the case shown below, an Arduino UNO was connected to the PC, and it’s using COM22, so COM22 will be used in the programming tool argument (replacing COM15 above).



So now we’ll change COM15 to COM22 in the External Tools Window



1. Tick the remaining boxes as shown above (don’t forget to tick Use Output Window) and press OK.
2. In the AVR menu, select Tools. You should see the Tool you added in the dropdown menu. Select it and it should program your Arduino over the USB cable, without needing an external programmer.



Notes:

1. You’ll need to change the path name if you downloaded the avrdude files to a different location than C:\avrdude\_7\_0.
2. Note also that the COM port in your case will almost certainly be different from COM15. When you connect Arduino to the USB port in your PC, this will be seen as a Virtual Serial COM port greater than COM1. You will need to change the line to reflect your set up. You can find the COM port number in the Device Manager on your PC.