

# Installing softwares for compiling code for and programming the OpenShoe system

John-Olof Nilsson and Isaac Skog

May 16, 2012

## **Abstract**

This documents gives installation instructions for the softwares necessary for compiling the code for the OpenShoe microcontroller and programming the same.

## **1 Introduction**

A number of softwares need to be installed in order to be able to compile code for the OpenShoe microcontroller as well as for programming the same. This document provides an overall installation instructions for these softwares. For detailed installation instructions please consult the documentation provided for the respective software.

There are many softwares and combination of softwares which can do the job of compiling the code and programming the microcontroller. However, here we give installation instructions for a set of programs for Windows which we have found the easiest to get up and running. Users with experience of microcontroller programming might prefer a different platform and set-up.

The main softwares/programs that need to be installed are

1. AVR Studio 5
2. AVR32 Studio 2.6
3. FLIP

In addition to installing these programs, the drivers for communicating with the microcontroller also need to be selected and system paths need to be set so that utility programs can be run from the command prompt.

Below installation instructions for the different softwares are given. Also comments and purposes of the different programs and instructions for checking out the code, setting up the IDE workspace and compiling the code are provided.

Screen shoots and Internet site paths are given on some occasions. Be aware that such screen shoots and paths might or might not correspond exactly to what you will see and the current location of things.

## 2 Programming development environment

The programming of the code for the microcontroller has been done in AVR Studio 5. However, many other AVR IDEs are available. The code is released as projects for the AVR Studio 5 and therefore, unless you are used to microcontroller programming, we recommend you to use AVR Studio 5. AVR Studio 5 is a free<sup>1</sup> stand-alone windows programming IDE provided by Atmel. The graphical interface of the IDE is shown in Figure 1. The program contain the necessary toolchain for compiling the code for the OpenShoe microcontroller.

The program can be downloaded from the Atmel homepage, [www.atmel.com](http://www.atmel.com). Follow the AVR Studio 5 link on the first page. Download an install according to provided instructions. The program will be dependent on other software components. Therefore depending on whats available in your system, the installer might automatically install Microsoft .NET Framework, Microsoft Visual Studio Shell, and AVR Jungo USB drivers.

## 3 Softwares for programming the MCU

For programming the microcontroller in the OpenShoe units, we have not used the AVR Studio 5 IDE. Instead we have found it the most convenient to program the system microcontroller directly with the command line tools `avr32program` and `batchisp`. Partially this have had to do with limited support in the mentioned IDE for the new AT32UC3C microcontroller used

---

<sup>1</sup>Registration required.

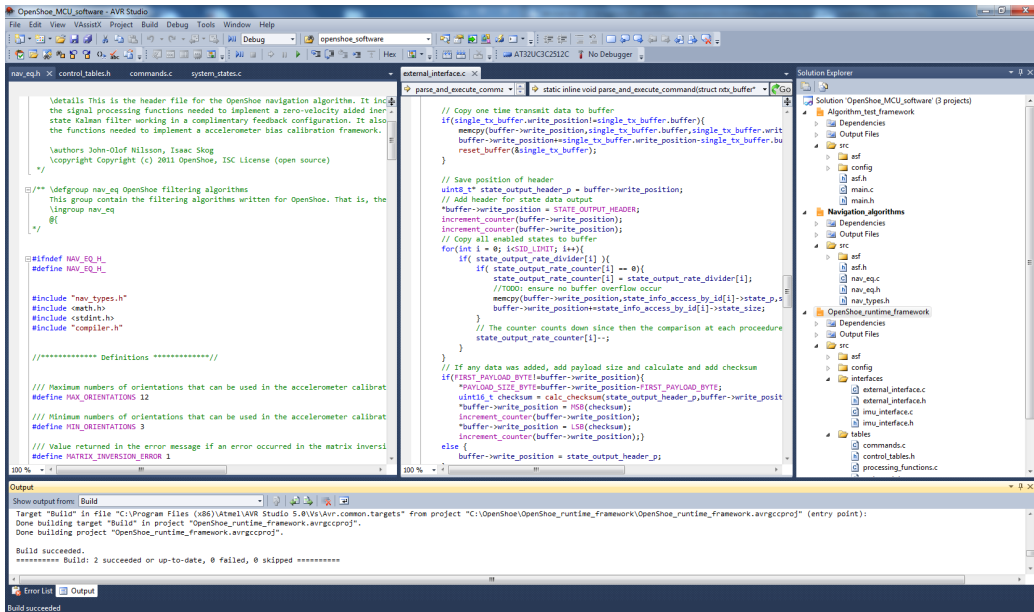


Figure 1: Graphical interface of the AVR Studio 5

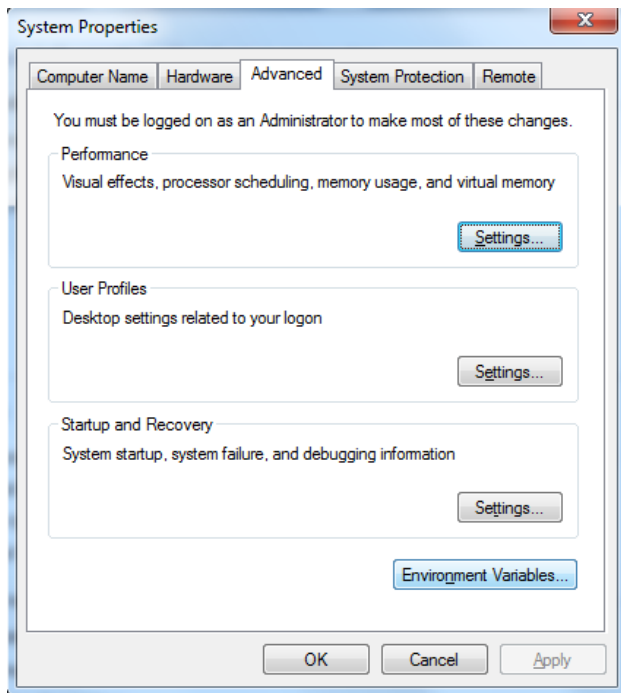
in the OpenShoe units. This might change with time. The `avr32program` tool is used to program the microcontroller via the JTAG interface while the `batchisp` tool is used to program the microcontroller via the USB interface.

The `avr32program` and `batchisp` tools are parts of, and can be accessed through another Atmel IDE, the AVR32 Studio 2.6, and the Atmel programming tool FLIP, respectively. We have found it easier to use them directly via the command prompt instead of via AVR32 Studio and FLIP. However, the easiest way of installing and getting the tools is by installing the “mother” programs. The programs can be downloaded from Atmel’s homepage.

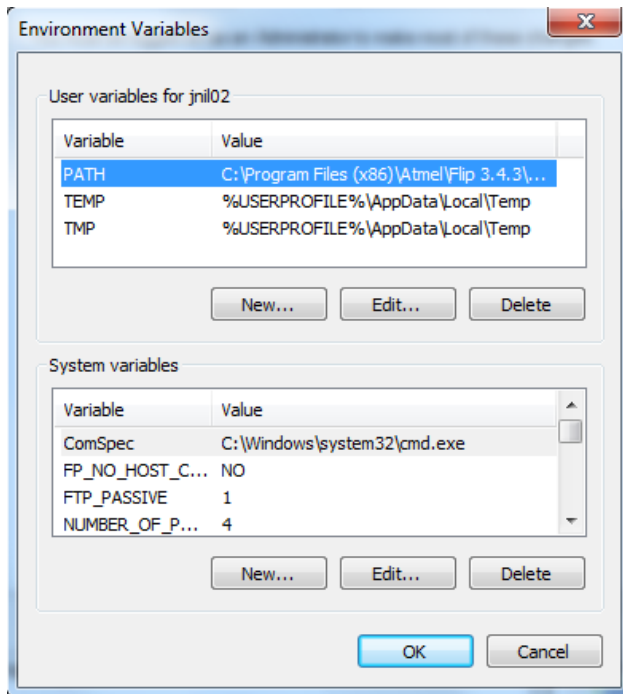
After installing the programs (AVR32 Studio and FLIP), the `avr32program` and `batchisp` tools can be found by searching for them in the program installation folders. Note that the current version of FLIP (3.4.3) does not support the AT32UC3C2512 micro controller but `batchisp` does. However, to use the programs in a convenient way some path variable need to be set. This is described below.

## 4 Setting path variables

Before the `avr32program` and `batchisp` programs can be used conveniently from the windows command prompt, the path variable need to be set such that they can be found without giving the full path. This is done by setting the windows PATH variable. The system path variable setting can be found under the control panel, system, and advanced system setting. As seen below, there a button **Environment Variables** can be found



Pressing the button brings up



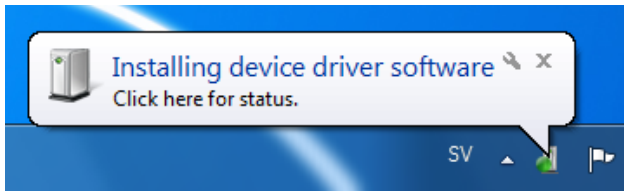
Edit the user variable `PATH` (create it if it's not there) to contain the full search path to the the `avr32program` and `batchisp` program folders. The different paths are separated by a semicolon. Press `OK`. The programs should be usable from the command prompt now. Any open command prompt will have to be restarted. The programs can be tested by the commands `avr32program -h` and `batchisp -h` respectively. If this does not work, the situation can be debugged by displaying all paths variables by typing `set` in the command prompt.

## 5 Hardware drivers

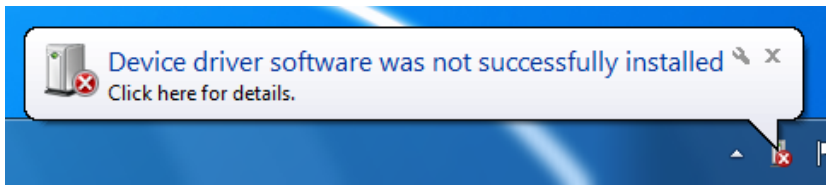
Before the microcontroller can be programmed via the bootloader and the USB port, drivers need to be selected for it. Pushing the connector button on the OpenShoe system (short circuiting pin 7 on the PCB) and connecting it to the USB port,<sup>2</sup> the first time one should see

---

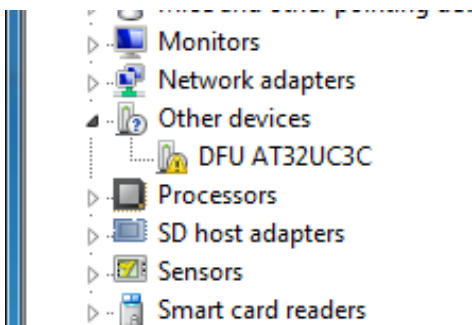
<sup>2</sup>The button only need to be pressed at the connection/power up movement.



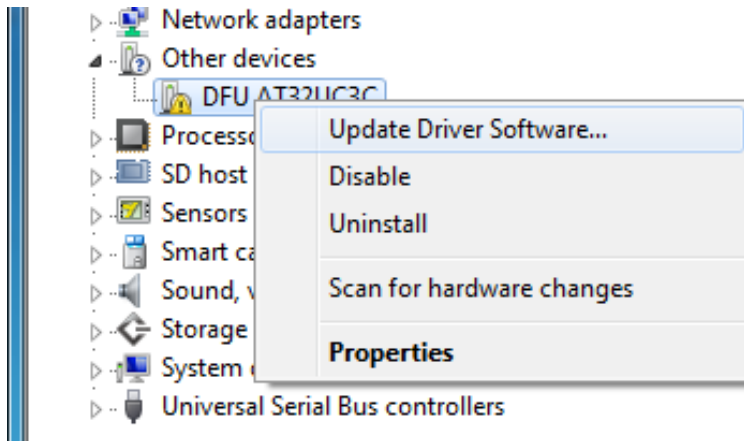
followed by



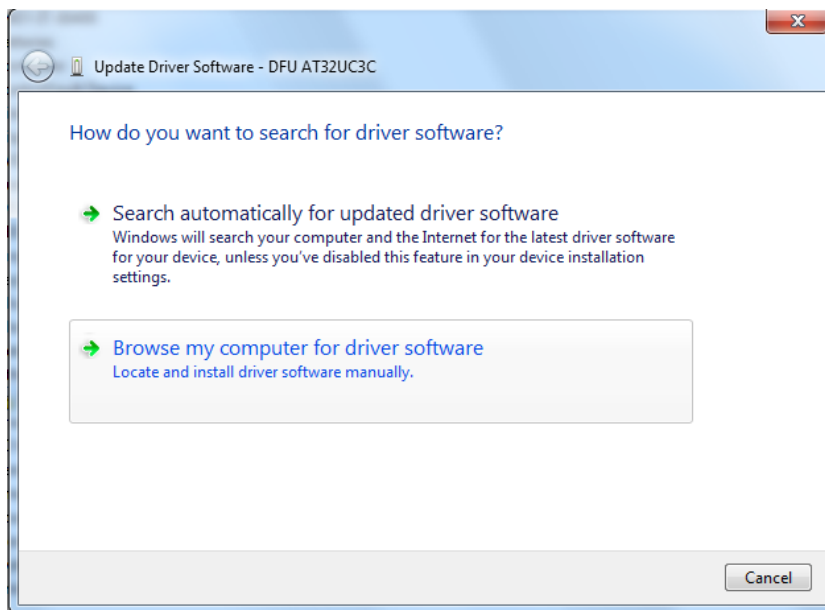
To install the hardware/select driver for the bootloader one needs to go to the device manager. There one should see



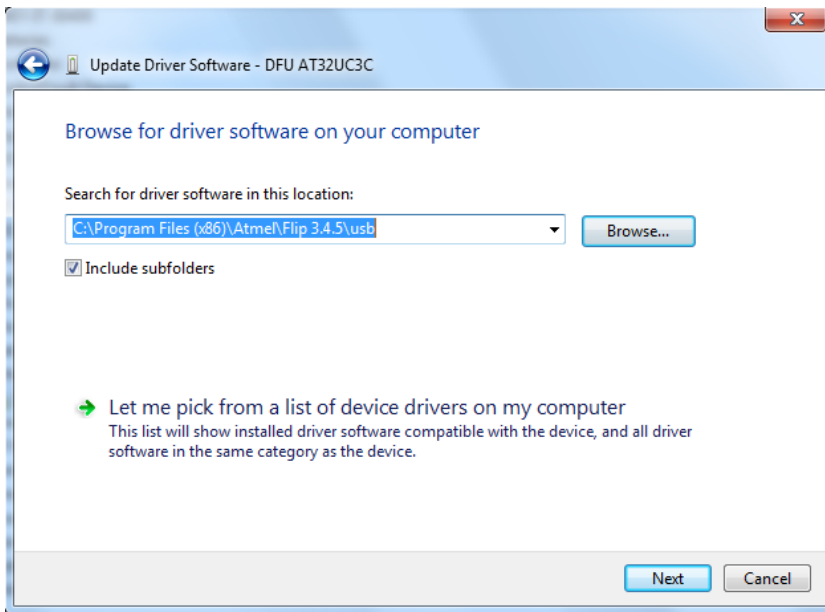
Right click on the device and select Update Driver Software....



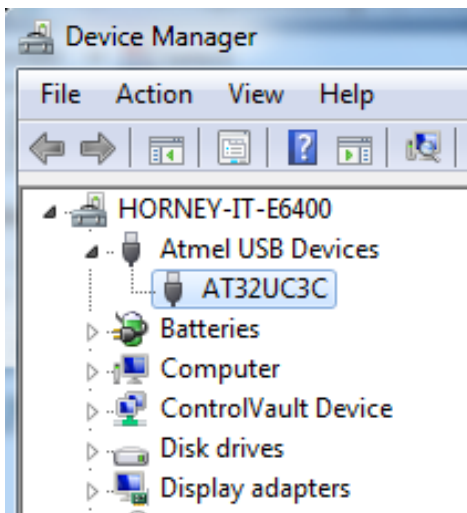
This will bring up a window like



Select the **Browse my computer for driver software** option. This will bring up a second window.



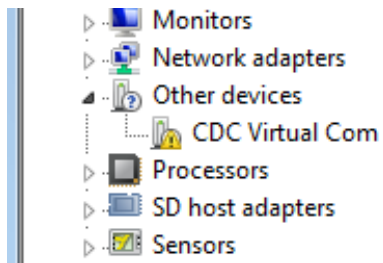
Type in the path to the `bin` folder under your FLIP installation. In this folder there should be a file `atmel_usb_dfu.inf`. Pressing next will install the driver for the bootloader. After this the unit should appear in the device manager as



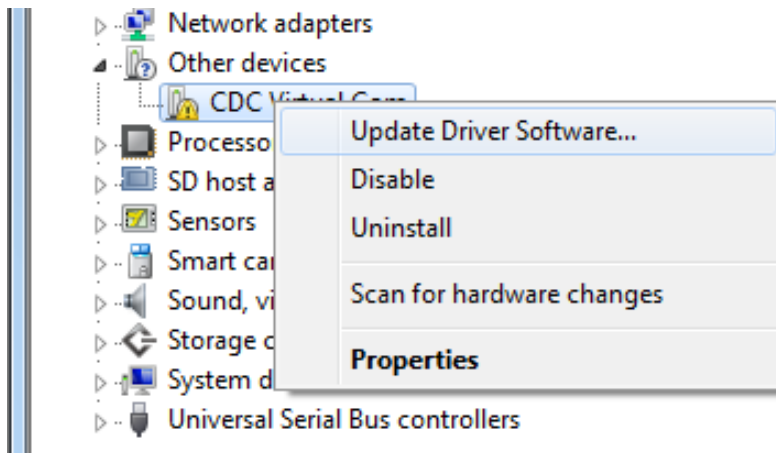
Similarly, after the OpenShoe microcontroller has been programmed properly, the device, now appearing as a virtual com port, need to be installed.



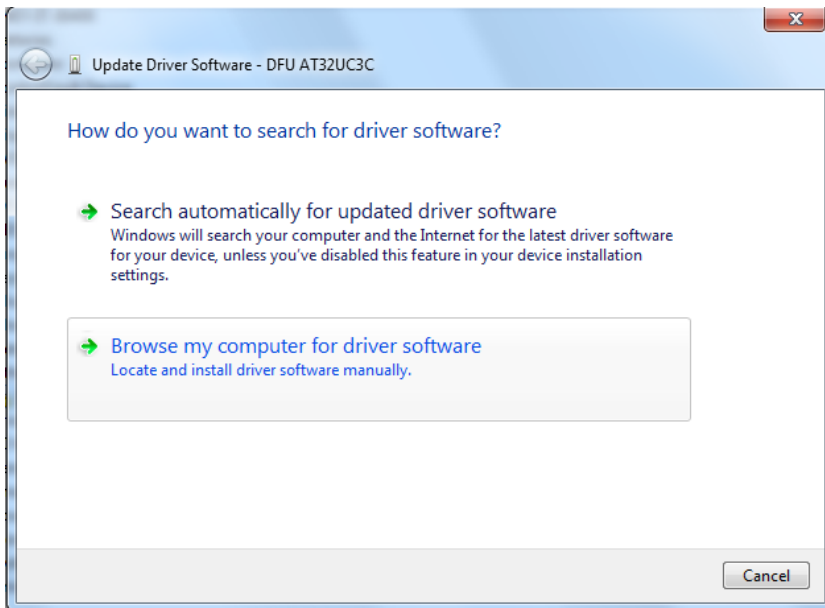
Directly after programming (or after plugging it in without pressing the connector button), the unit should appear in the device manager as



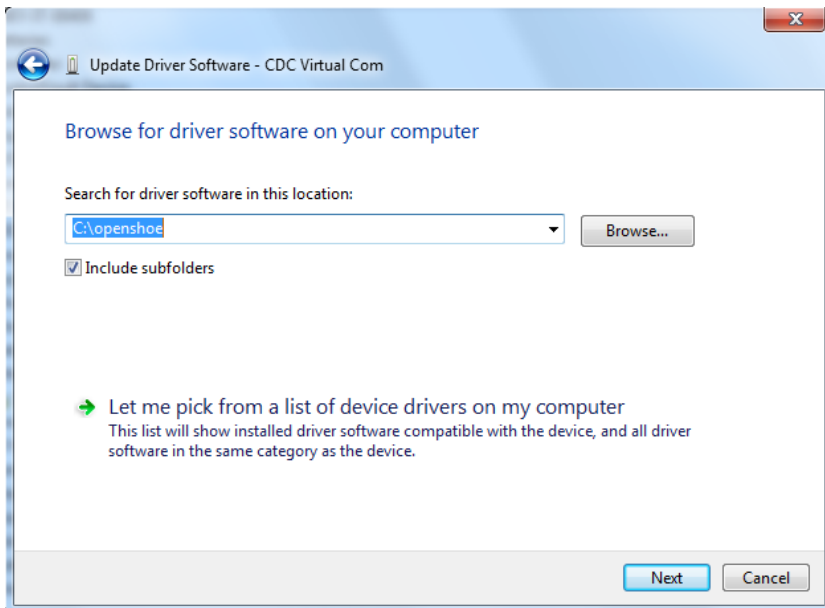
Right click on it and select Update Driver Software....



This will bring up the same window as before

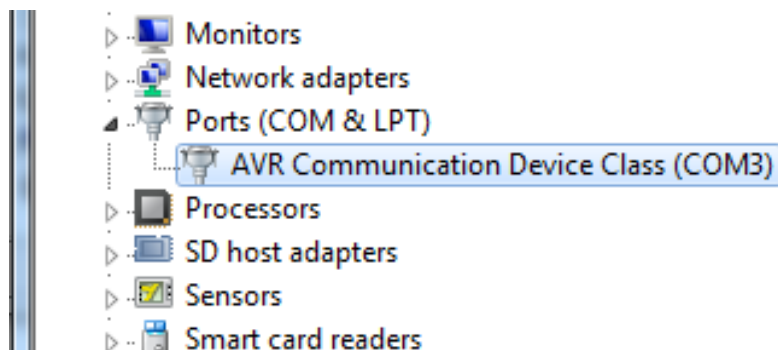


Select the **Browse my computer for driver software** option. This will bring up a second window.



This time you should select the folder in which you downloaded the boot-

loader tools. It should contain a file `avr_cdc.inf`. Pressing next and the driver should be installed. Following this the device should appear as



in the device manager. Note that a different COM-number might be assigned to it. After this the unit can be used as a virtual com port and controlled by the Matlab control scripts.

## 6 Installing git and downloading the code

The most recent code of the OpenShoe project is distributed via git repositories on SourceForge. Releases are distributed as zip-archives on [http://www.openshoe.org/?page\\_id=28](http://www.openshoe.org/?page_id=28). For most users the zip-archives will most probably be the best alternative. They can be directly downloaded and unzipped. If you want to use the latest code you need to install git and check out the code from the repositories. Git can be downloaded from <http://git-scm.com/>. Once git is installed you can check out the OpenShoe-runtime-framework code by the commands

```
git clone git://git.code.sf.net/p/openshoe/framework
code_destination
```

where `code_destination` is the path to the folder in which you want the code to end up. The algorithm implementation can be checked out by

```
git clone git://git.code.sf.net/p/openshoe/algorithms
code_destination
```

The other code repositories can be checked out similarly.

The code that is checked out contain AVR Studio 5 project files. The easiest way to compile the code is to open AVR Studio 5, create a new solution and add these project. The `Navigation algorithm` project should be possible to build as is. To build the `OpenShoe runtime framework` project, the pathes specifying where to look for the `Navigation algorithm` project header and library files need to be set under the project properties. Once this is done, the project should be possible to compile. This should give a `OpenShoe_runtime_framework.elf` file under the `Debug` or `Release` folder. This is the file which should be programmed to the system microcontroller as described in the microcontroller programming instructions found at [http://www.openshoe.org/?page\\_id=40](http://www.openshoe.org/?page_id=40).