Assembly Language Programming With Atmega8 And Microchip Studio

The fundamental concepts associated with assembly language programming using the Atmega8 microcontroller are discussed. Topics include introduction to Microchip Studio 6 and assembly language programming.

4.1 Introduction to Microchip Studio

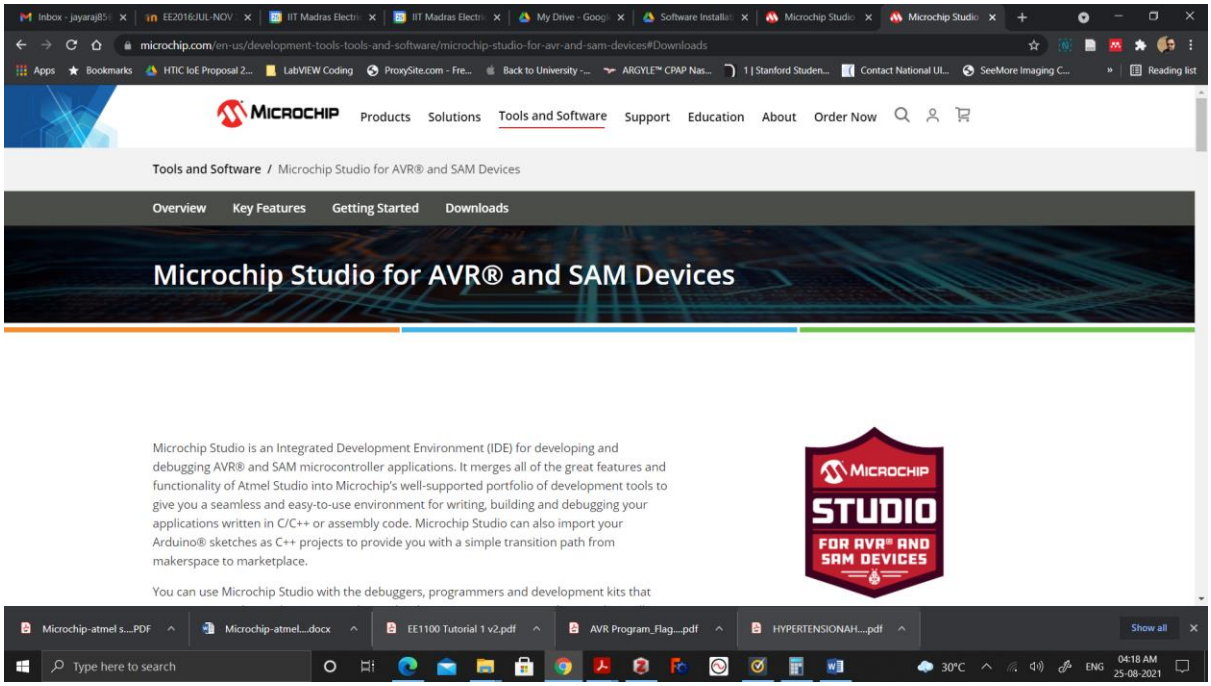
Microchip Studio is a sophisticated free development environment **(Windows only)**. It provides everything needed to seamlessly edit, build, download and debug applications on Atmel microcontrollers. Microchip Studio also manages and provides a direct interface to the Atmel software framework, and is able to directly access the Atmel Gallery. Atmel start generates pre-configured and build-ready Microchip Studio projects (Atmel START also support IAR, Keil and makefile projects).

4.1.1 Downloading and installing Microchip Studio 7

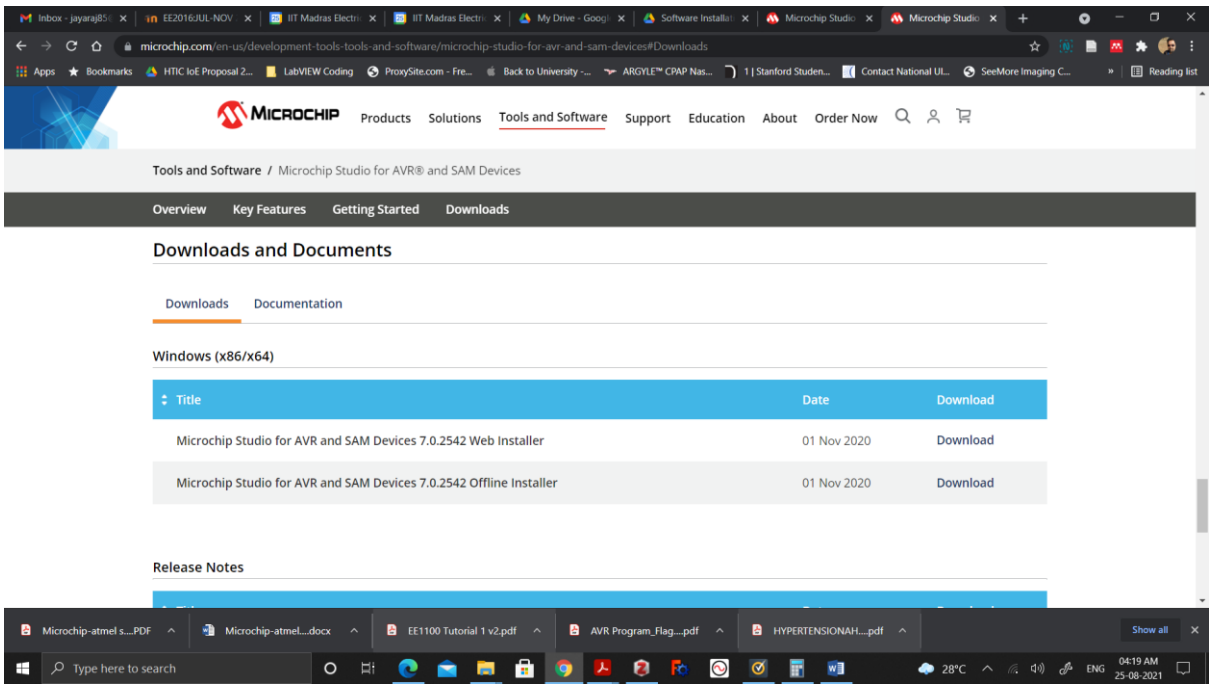
Download the Microchip Studio 7 installer for Windows from the link below. After downloading, open the file. Please be aware that the installation process will take around 30-40 minutes.

Link: You can find the latest installer for Microchip Studio from the company website in the following page.

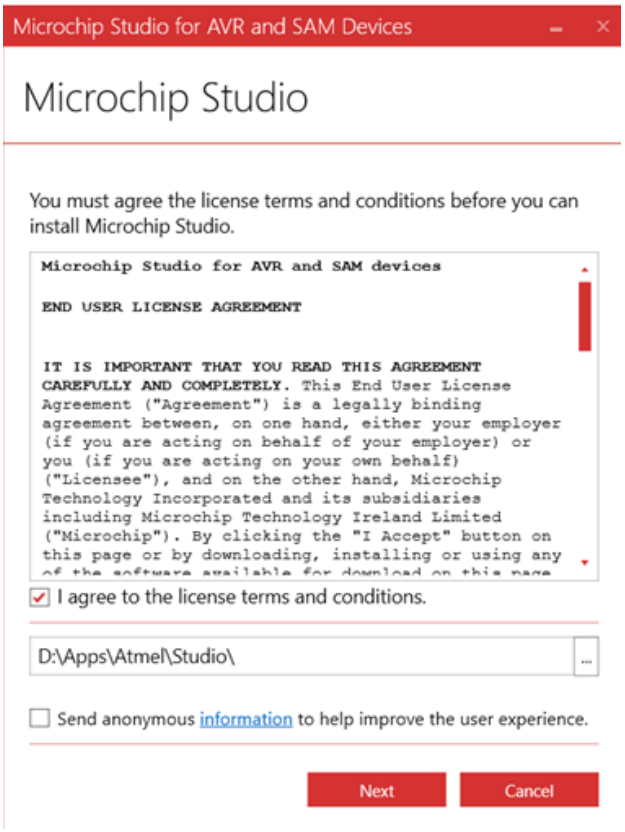
https://www.microchip.com/en-us/development-tools-tools-and-software/microchip-studio-for-avr-and-sam-devices#Downloads



Click on the Downloads Tab and it will take you to the portion of the web-page where the installers are available.



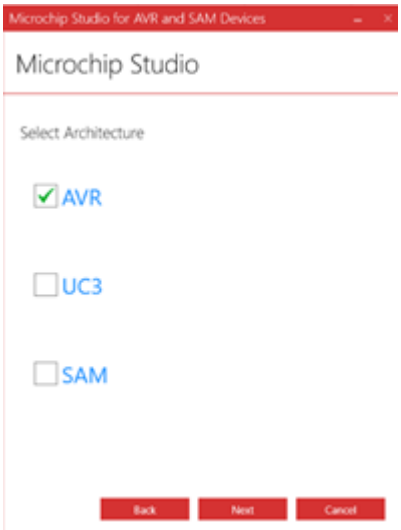
Choose offline installer and download the full 933 MB installer file. Once the installer file is downloaded completely, double click on the as-installer-7.0.2542-full.exe file to start the installation.



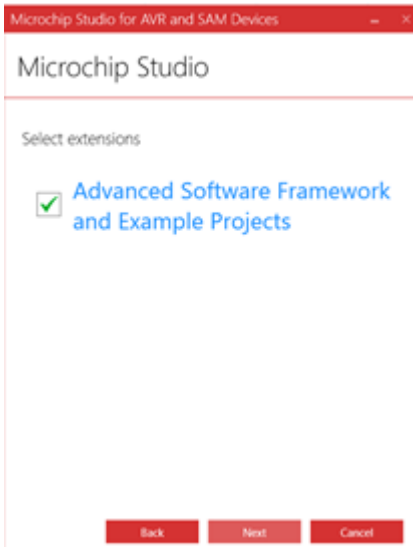
You have to agree to the license to continue, sending anonymised data to Atmel is up to you.

NOTE: You can change the default installation directory BUT make sure you create folders named “Atmel” and “Studio” inside it at your file location and manually add backslash “\” at the end as shown above to avoid installation errors in the later stages.

After clicking Next you will reach the below figure, select AVR only as it is enough for us. You can install all the 3 if you wish. Click Next again.



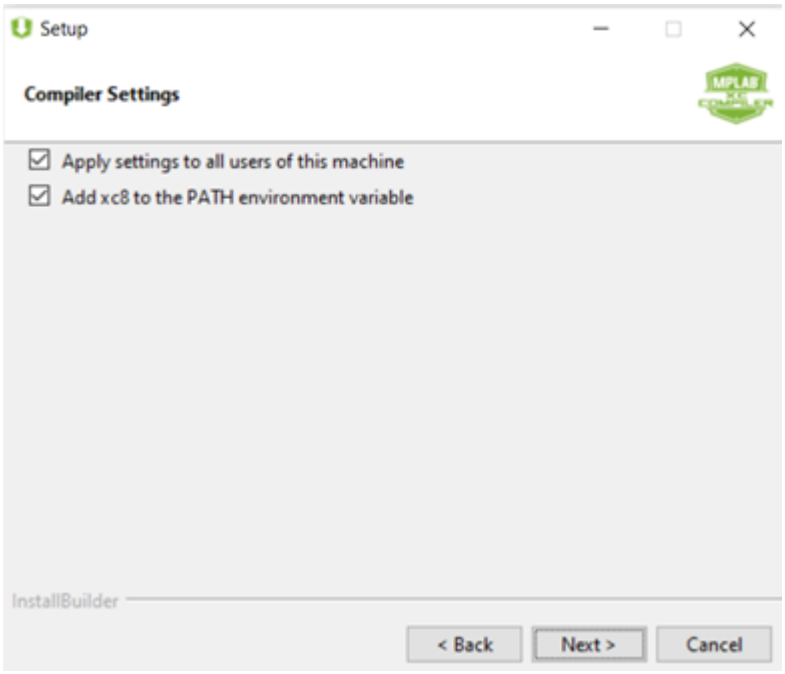
You might want to keep this option selected. After this keep on clicking Next when prompted which will eventually start the installation process.



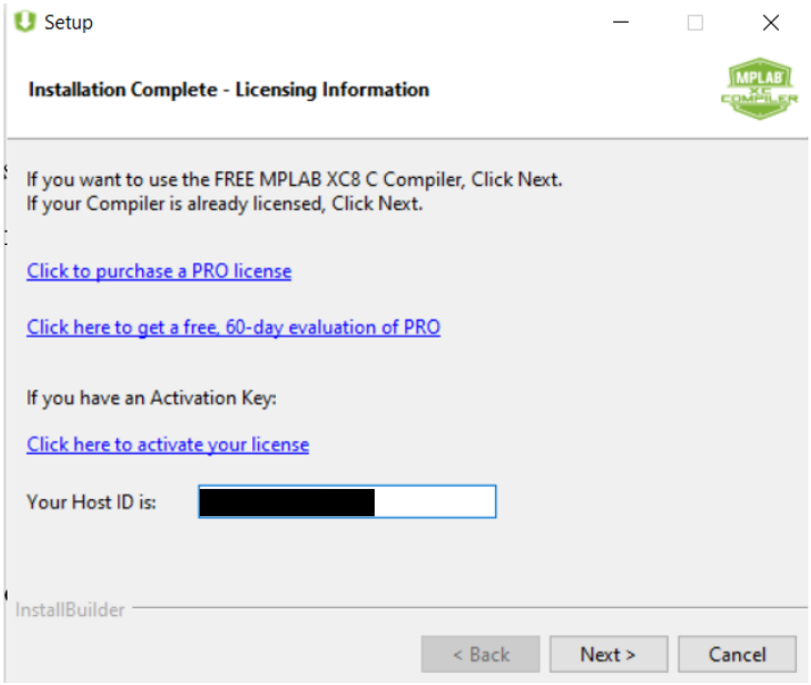
During the installation, this popup window (see below picture) will come, keep clicking Next to go on.

NOTE: Here also you can change the installation directory as before. Again make sure the destination folder is named the same as the default one (Illustration NOT shown here).

You might want to select both the options here.

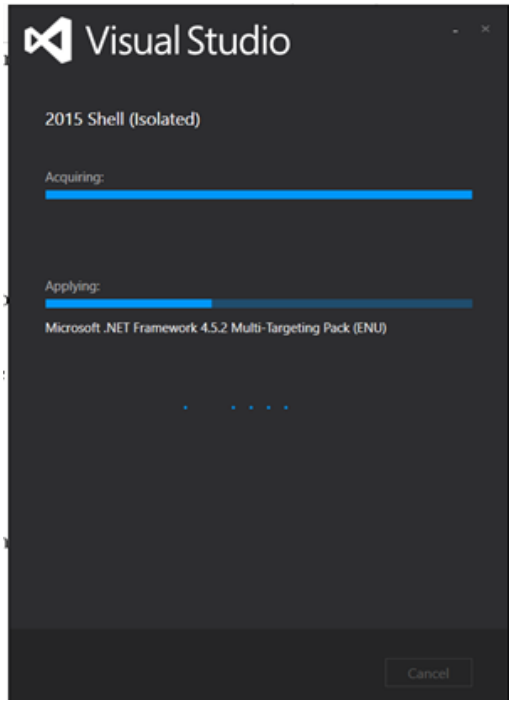


Below prompt will appear upon completion of Compiler installation, Studio installation is still remaining though!



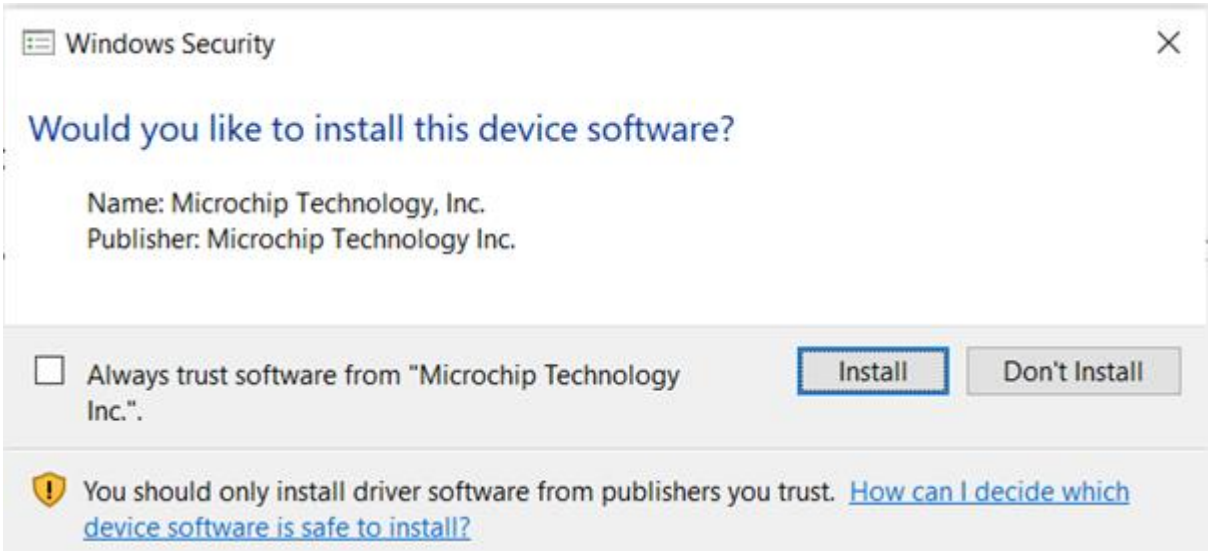
Click on Finish to continue with the Studio installation.

Another popup window like in the below picture will show up, you DO NOT have to click anything here, it will automatically get installed.



The below picture shows one of the many similar prompts that will pop up in the final stages of the installation. You can select the “Always Trust software” checkbox and click Install to proceed further each time.

NOTE: Though it is advised to NOT select the “Always Trust software” option, we have done that. If you don’t do so then you will get too many prompts and each time you have to click Install. Selecting this option is your call.

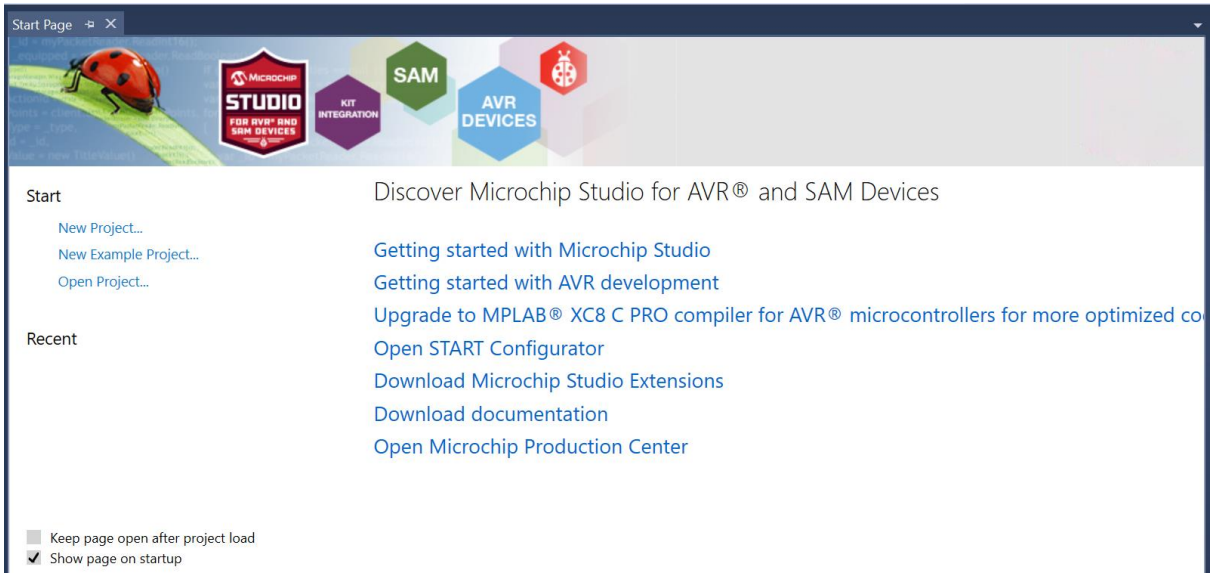


The below picture appears upon the successful completion of the installation process. You can select the checkbox to directly open the freshly installed Microchip Studio.

Click Close to COMPLETE THE INSTALLATION PROCESS.

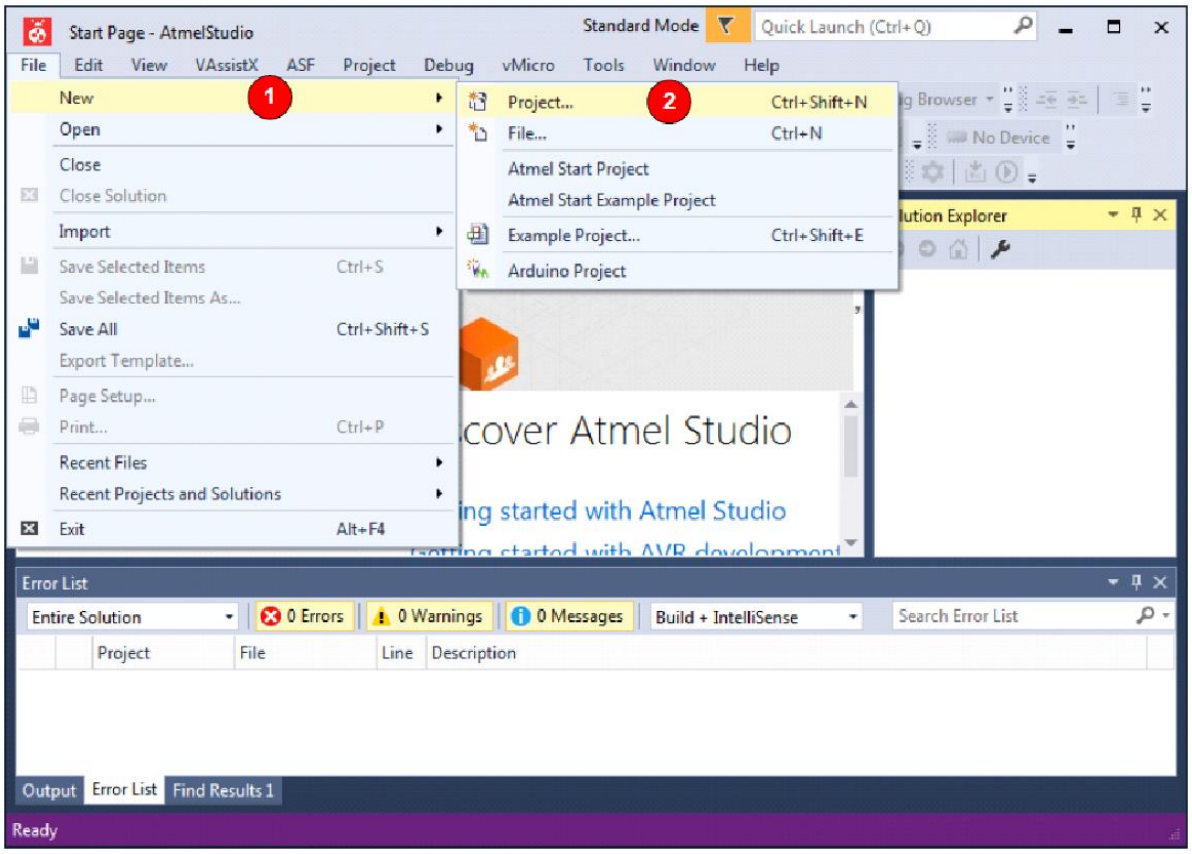
4.1.2 Opening Microchip Studio

Go to the ***Start*** menu and open Microchip Studio.



4.1.3 Creating the first project

1. Go to the ***File*** menu. Choose ***New*** and then ***Project***



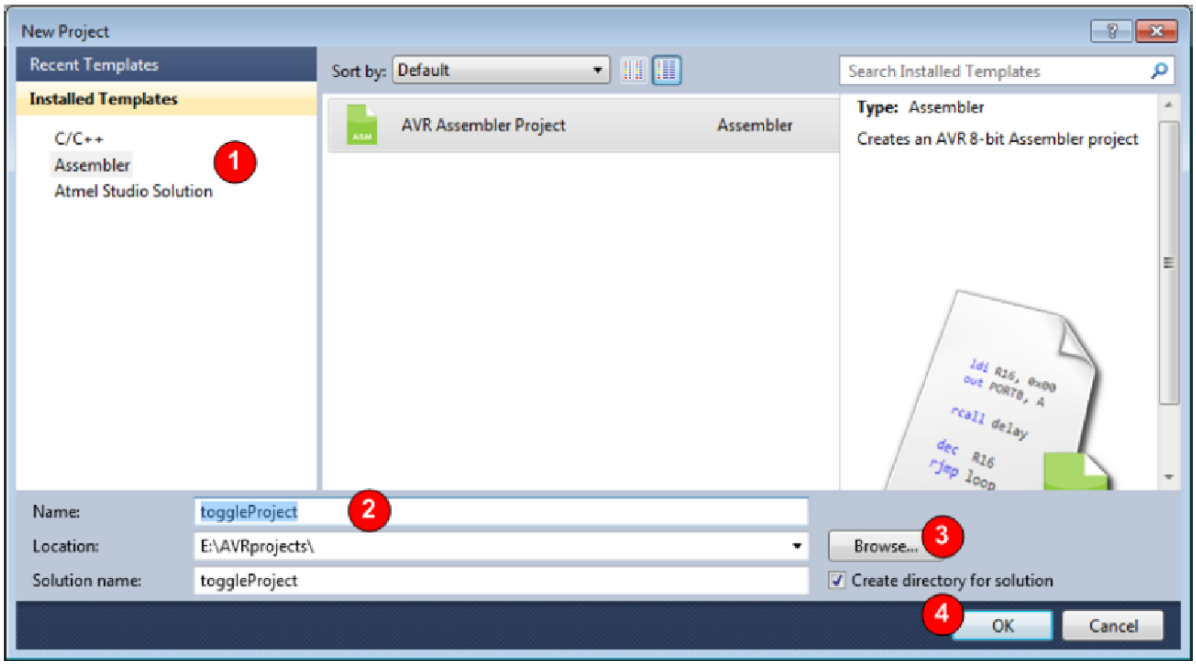
2. In the opened dialog,

a. Choose ***Assembler***.

b. Name the project as ***example1***.

c. Choose the path where you like to save the project by clicking on the ***Browse*** button.

d. Press OK.

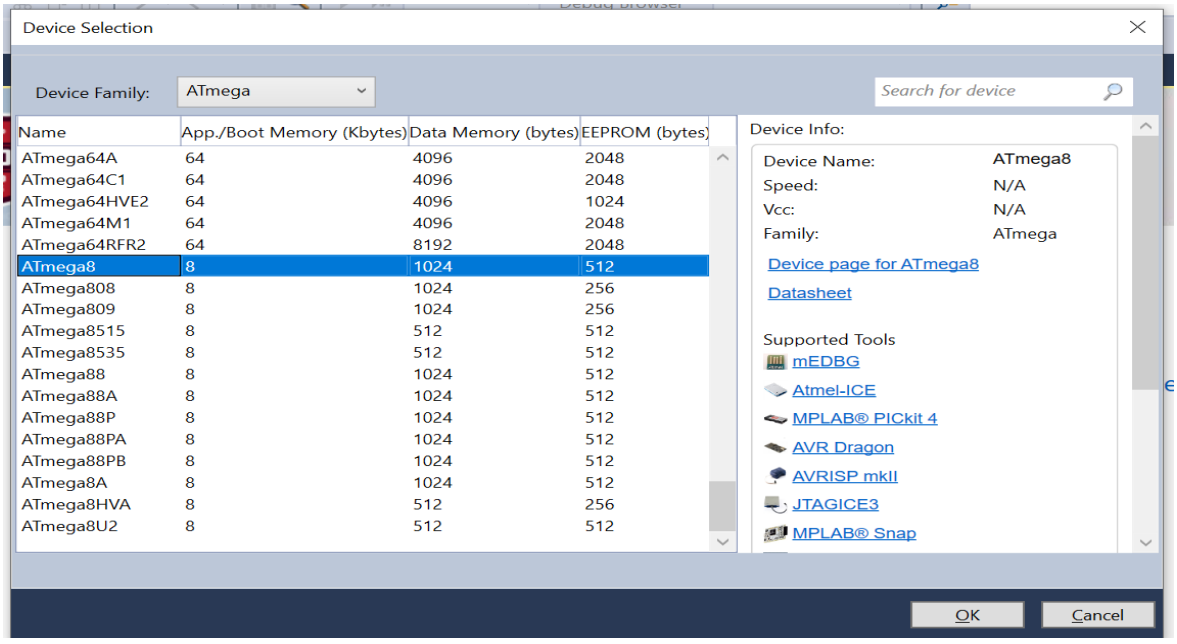


3. In the ***Device Selection*** dialog

a. Select ***ATmega*** as the ***Device family***.

b. Choose ***ATmega8*** (or any other Chips you want to use)

c. Select OK.

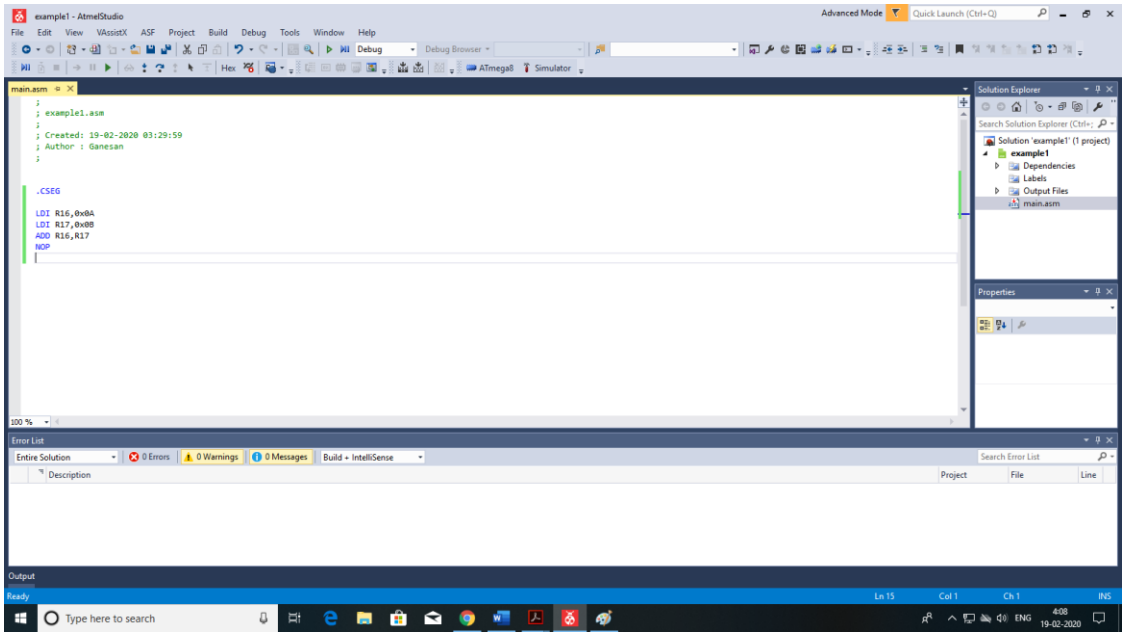


The compiler automatically makes the project ***assembler1*** and adds an assembly file to it.

4. Writing the first assembly program

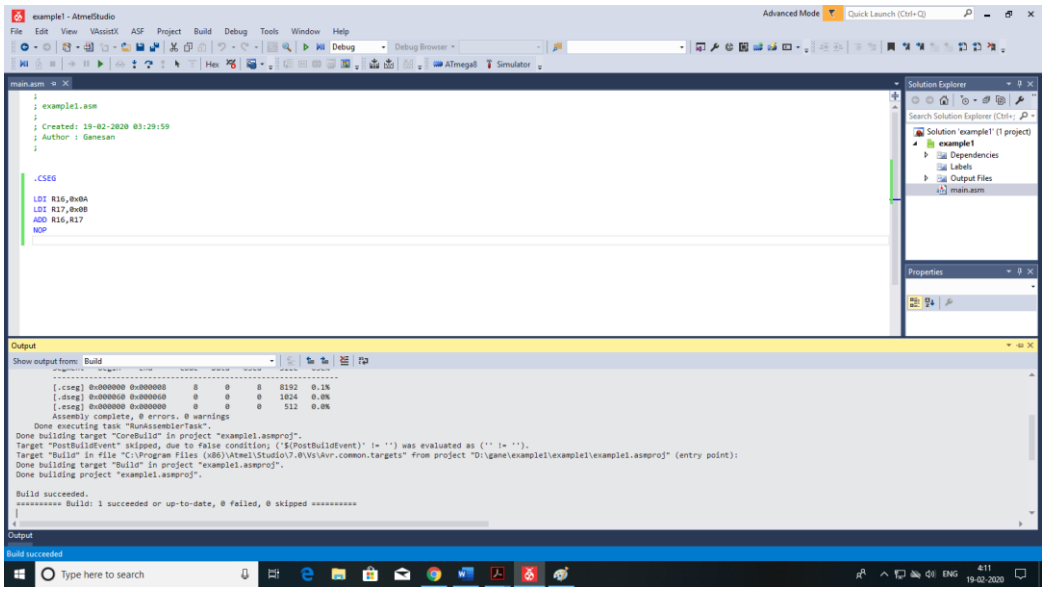
The following program adds two hexadecimal numbers.

```
.CSEG
LDI R16, $14
LDI R17, $15
ADD R16, R17
NOP
```



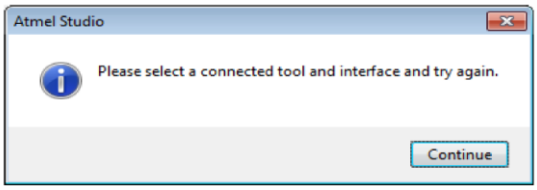
Building

Press *F7* to assemble, or choose *Build Solution* from the *Build* menu. The results of assembling the program are shown in the *Output* window.

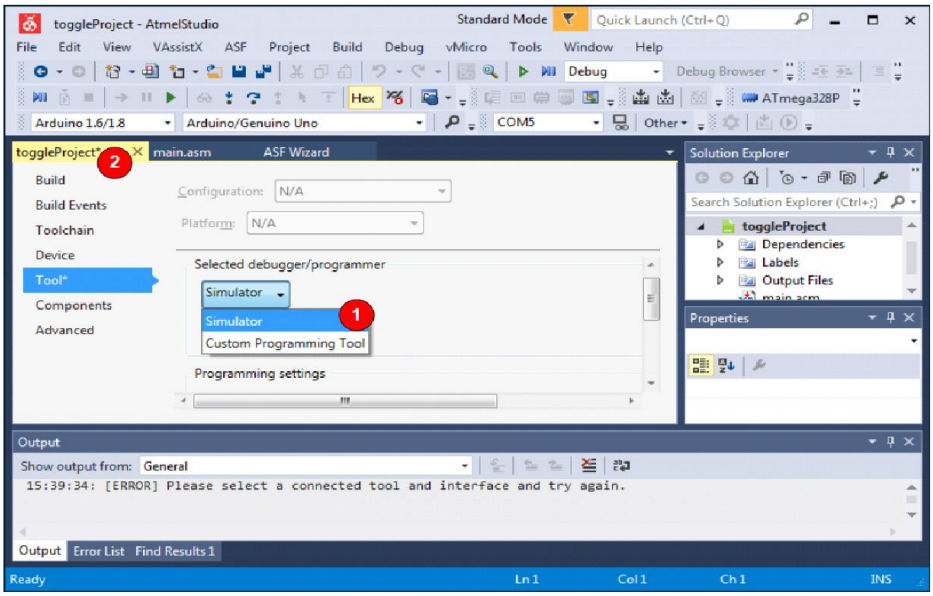


Debugging

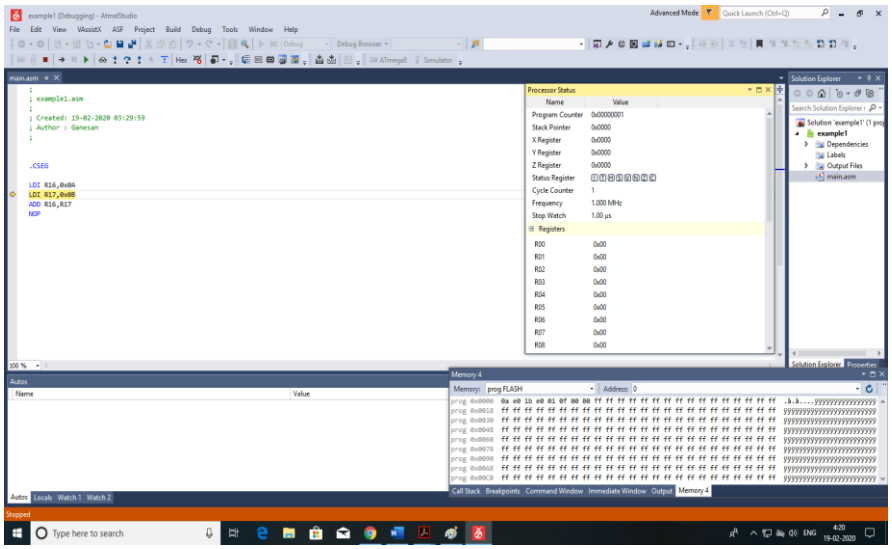
1. To start debugging, press ***Alt+F5*** or choose ***Start Debugging and Break*** from the ***Debug*** menu.
2. The following Dialog appears and asks you to select the debugging tool. Press ***Continue***.

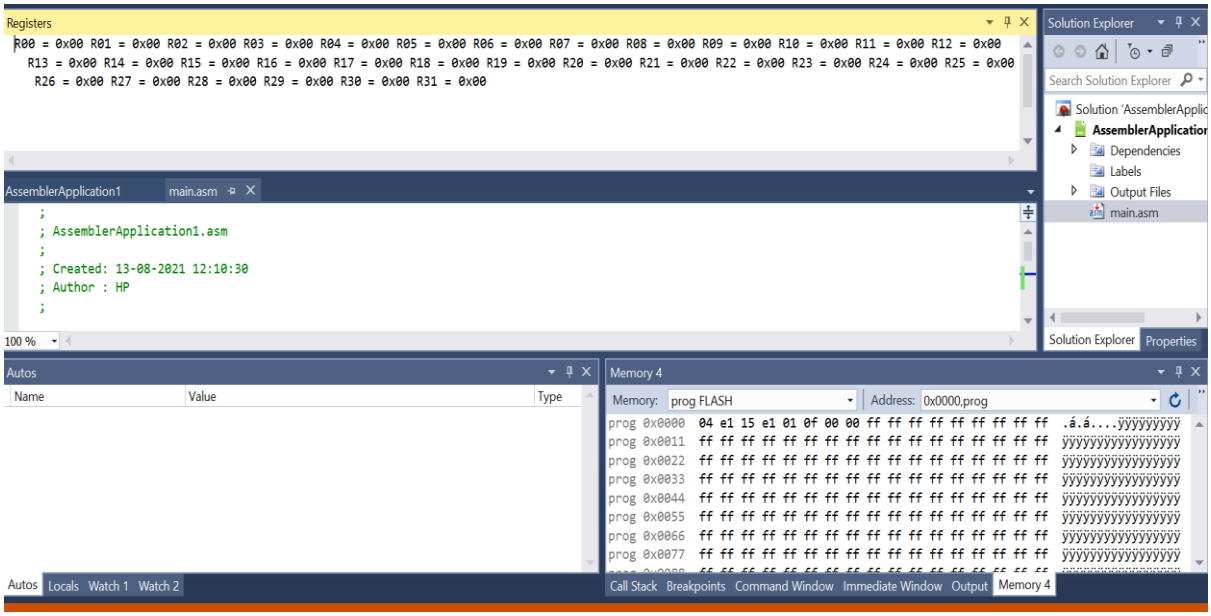


3. In the following window, choose ***Simulator*** as the debugger and save the file (Ctrl+S), and then close the tab by clicking the x next to ***assembler1***.



4. Press ***Alt+F5*** again. Now a yellow cursor is on the first line of the main program and the IDE is ready to debug the program.





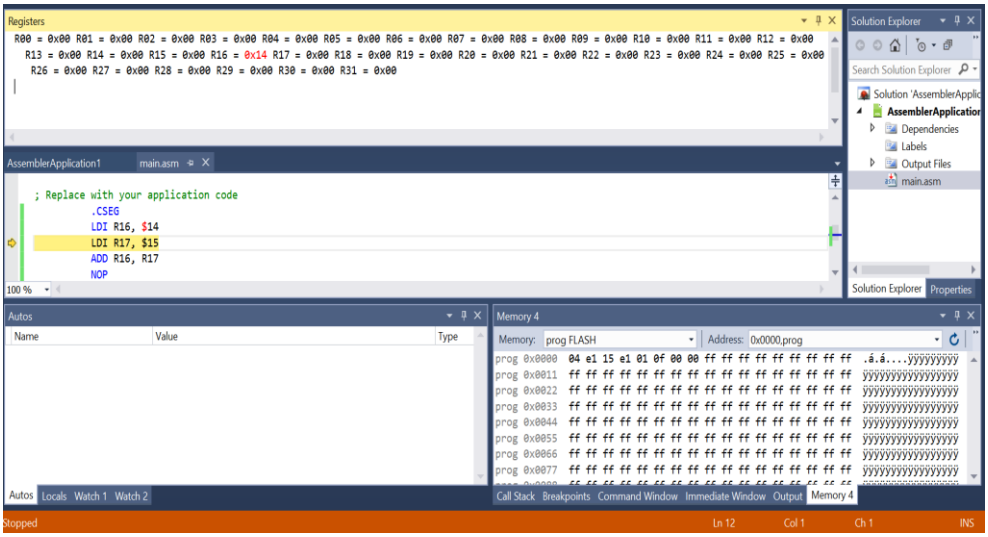
To view the registers, press Alt+5. There are R00 to R31 registers – 32 registers as studied in theory sessions.
All of them are initialized to ZERO.

After execution of the program, the corresponding registers are configured to values as indicated in the program.

5. To execute the instructions line by line press ***F10*** or click on the ***Step over*** icon

Step Into vs. Step Over

Both ***F10 (Step over)*** and ***F11 (Step into)*** execute one instruction and go to the next instruction. But they work differently when the cursor is on a function call. If the cursor is on the function call, ***Step into*** goes into the first instruction of the function, but ***Step Over*** executes the whole function and goes to the next instruction.



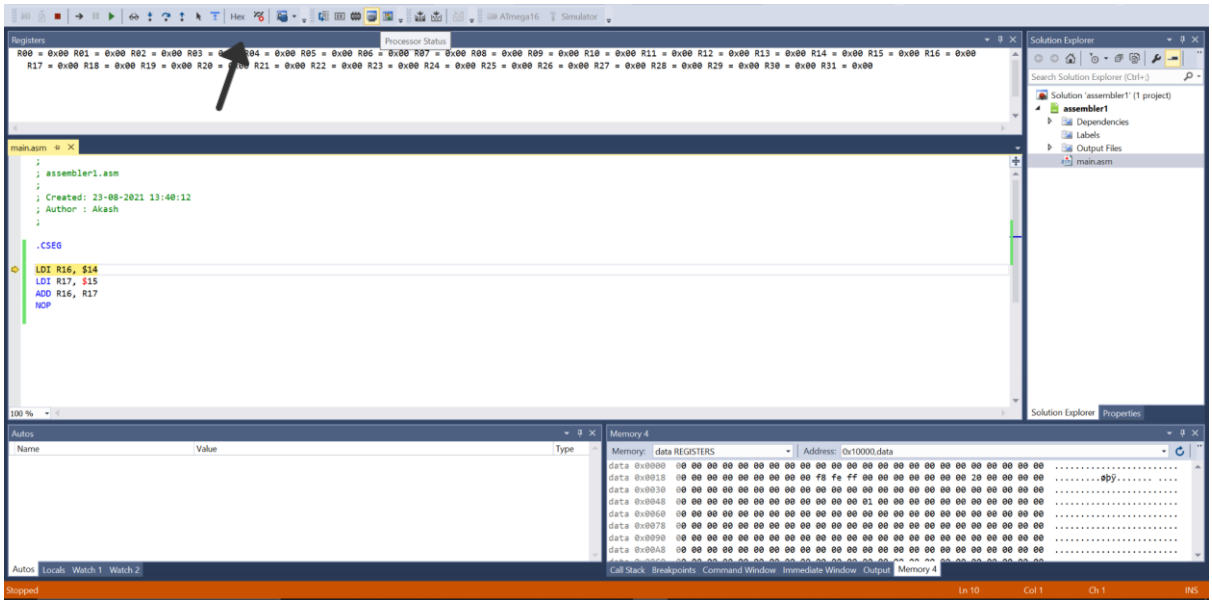
Step Out
If the execution is in a function, you can execute the function to the end by pressing the ***Step Out***.

Run to Cursor
You can put the cursor on an instruction and then press the Run to Cursor button. In the case, the program runs until it reaches the instruction which the cursor is on it.

Processor Tab
The Processor tab shows the current values of the CPU registers including R0-R31, SP (Stack Pointer) and PC (Program Counter). You can also change the values of registers by double clicking on their values and typing a new value.

Enabling Processor Tab

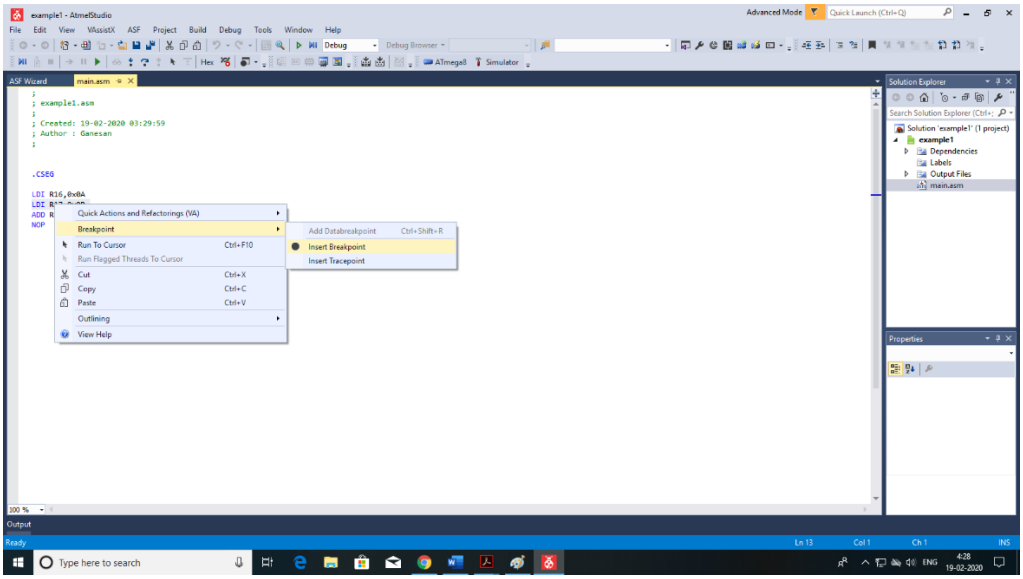
To view the Processor Tab, click the “Processor Status” button in the toolbar.



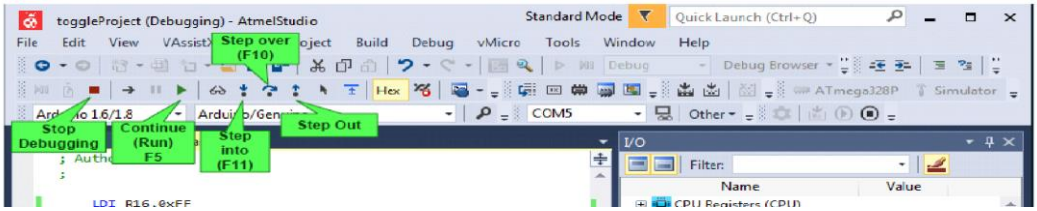
Using Breakpoints

If you want to debug a portion of a program, add a breakpoint to the beginning of this part of the code and press the run button (or F5). The IDE runs the program and when it reaches the breakpoint, it stops running and the yellow cursor is shown on the breakpoint line. Below, you see the steps in detail.

1. Right click on the "LDI R17,0x0A" instruction. A pop-up menu appears. Choose ***Breakpoint*** and then ***Insert Breakpoint***. A red bullet appears on the left side of the " LDI R17,0x0A " instruction.



2. Press ***F5*** or the ***Continue*** button. The IDE runs program until it reaches the Breakpoint. Now, you can continue debugging from the breakpoint as usual - using the ***Step into*** and ***Step over*** buttons.



3. Using ***Stop Debugging (or Ctrl+Shift+F5)***, you can stop debugging whenever you want.

4. Right click on the same instruction and choose ***Breakpoint → Delete Breakpoint*** to remove it.

Some useful resources:

Try to download the user manuals and application notes and similar support documents for the Microchip Studio and AVR Atmega 8/16/32 series of controllers and read-up to have better understanding of the chip hardware organisation and instruction set architecture.

A User-Guide to Microchip Studio is available under the Documentation tab of the page : https://www.microchip.com/en-us/development-tools-tools-and-software/microchip-studio-for-avr-and-sam-devices#Downloads

Note : You may have to create a free account at the site to get access to the User Guide.