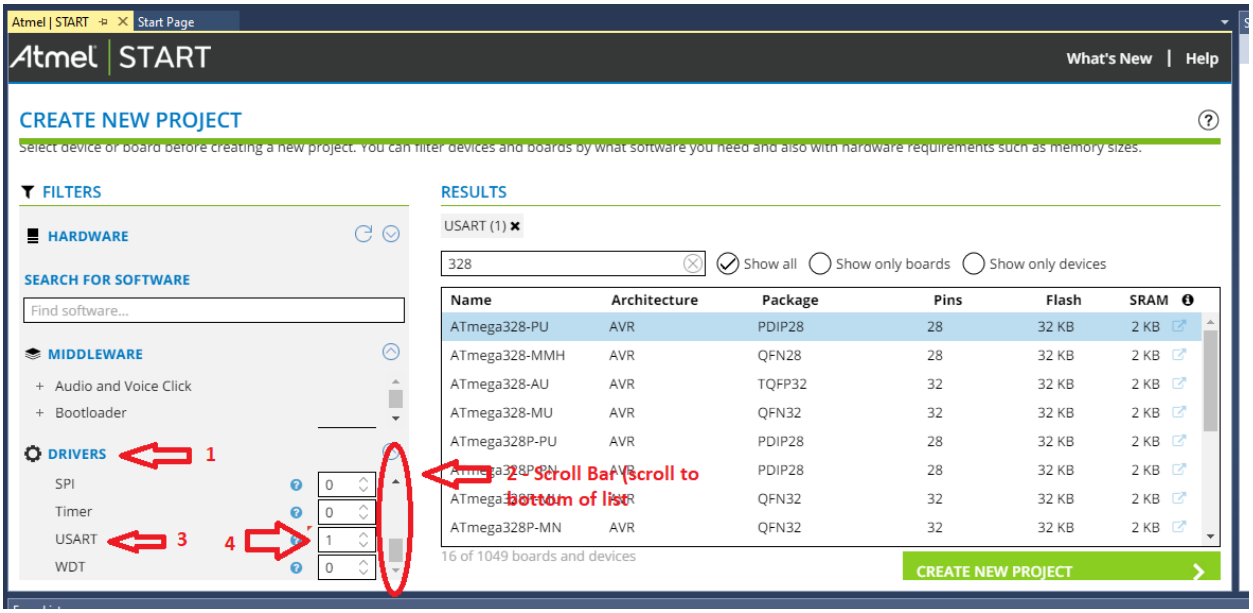
TECH 3233
Setting Up USART for printf

To set up ATMEL studio to use printf, start the project as you did in LAB3 procedure Step 1, but in addition to selecting the ATmega328-PU, you will need to add the USART. To do this: Find Drivers (1 below) and use the scroll bar (2) to find USART(3), in the box next to it (4), put in a value of 1.



CREATE NEW PROJECT

Select device or board before creating a new project. You can filter devices and boards by what software you need and also with hardware requirements such as memory sizes.

FILTERS

HARDWARE

SEARCH FOR SOFTWARE

Find software...

MIDDLEWARE

- + Audio and Voice Click
- + Bootloader

DRIVERS

SPI	0
Timer	0
USART	1
WDT	0

RESULTS

USART (1) ✖

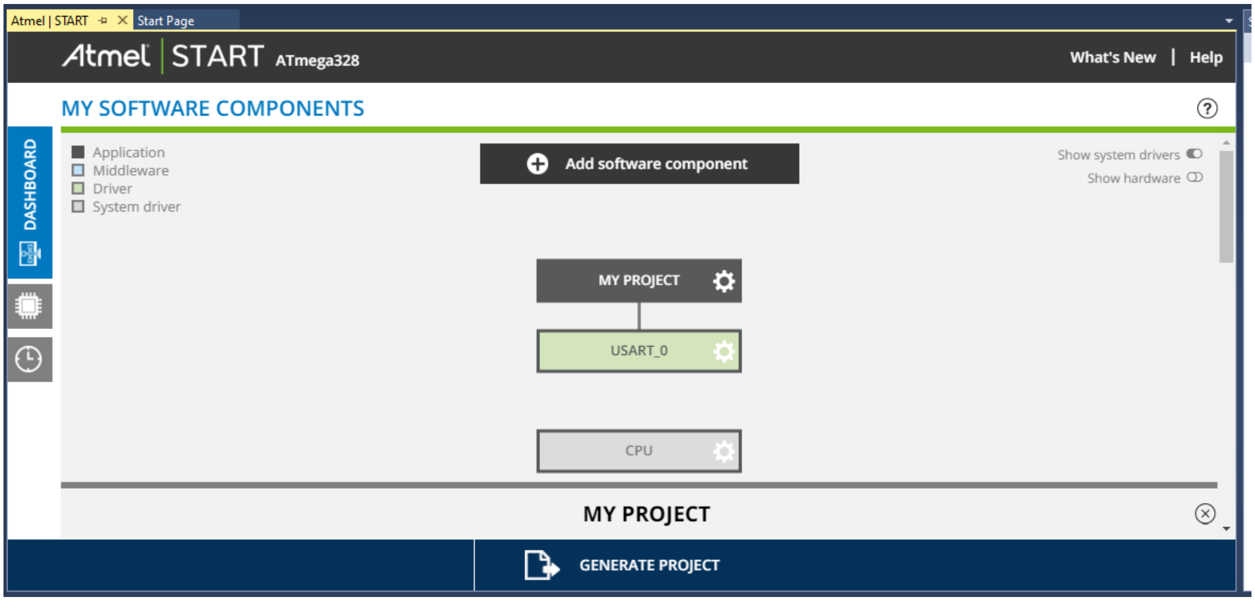
328 ☑ Show all ○ Show only boards ○ Show only devices

Name	Architecture	Package	Pins	Flash	SRAM
ATmega328-PU	AVR	PDIP28	28	32 KB	2 KB
ATmega328-MMH	AVR	QFN28	28	32 KB	2 KB
ATmega328-AU	AVR	TQFP32	32	32 KB	2 KB
ATmega328-MU	AVR	QFN32	32	32 KB	2 KB
ATmega328P-PU	AVR	PDIP28	28	32 KB	2 KB
ATmega328P-PN	AVR	PDIP28	28	32 KB	2 KB
ATmega328P-MU	AVR	QFN32	32	32 KB	2 KB
ATmega328P-MN	AVR	QFN32	32	32 KB	2 KB

16 of 1049 boards and devices

Then hit “Create Project”.

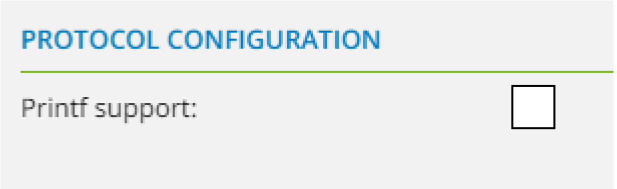
You should have:



MY SOFTWARE COMPONENTS

MY PROJECT

Change the CPU clock speed to 16MHz as you have in the past, but now also click on USART_0 and scroll down to find:



PROTOCOL CONFIGURATION

Printf support: □

And put a check in the box (no other changes are required). Now hit “Generate Project”.

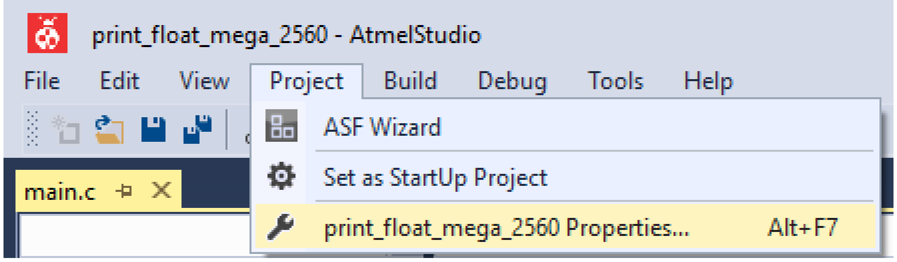
This will allow you to use printf in your program.

Skip this next part if you are NOT USING PRINTF to print Floating Point Values

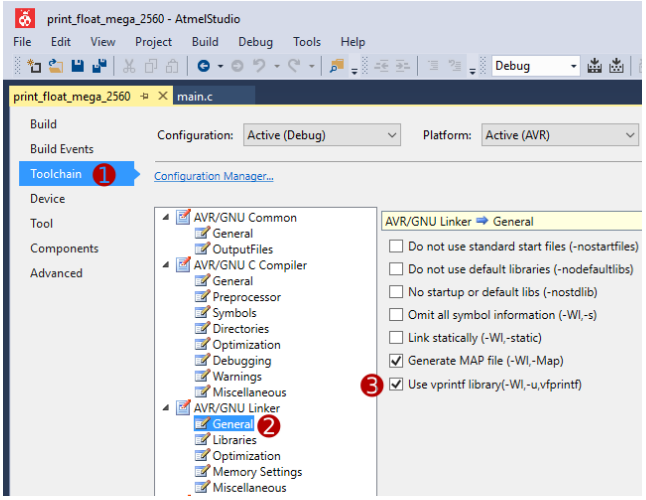
If you wish to print **floating point number**, you MUST tell the compiler to enable the floating point printf (this is NOT included by default to save memory space).

To add floating point capabilities you need to do the following (from https://startingelectronics.org/articles/atmel-AVR-8-bit/print-float-atmel-studio-7/)

> In Atmel Studio 7 on the top menu, click **Project → <project name> Properties...** to bring up the properties page for the currently open project. The image below shows the menu in Atmel Studio 7 for a project named **print_float_mega_2560**.



> Click **Toolchain** in the page at the left of the project properties page and then **General** under the **AVR/GNU Linker** item as shown in the image below. Finally check the **Use vprintf library(-Wl,-u,vfprintf)**.



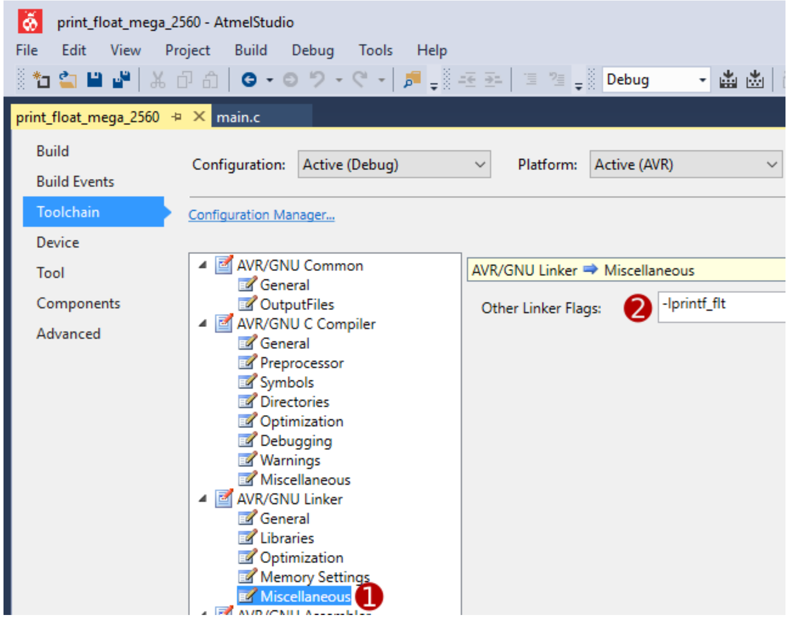
Configuration: Active (Debug)
Platform: Active (AVR)

AVR/GNU Linker ➡ General

☐ Do not use standard start files (-nostartfiles)
☐ Do not use default libraries (-nodefaultlibs)
☐ No startup or default libs (-nostdlib)
☐ Omit all symbol information (-Wl,-s)
☐ Link statically (-Wl,-static)
☑ Generate MAP file (-Wl,-Map)
☑ Use vprintf library(-Wl,-u,vfprintf)

Now click **Miscellaneous** under the **AVR/GNU Linker** item and add the following in the **Other Linker Flags** box as shown in the following image.

```
-lprintf_flt
```



Configuration: Active (Debug)
Platform: Active (AVR)

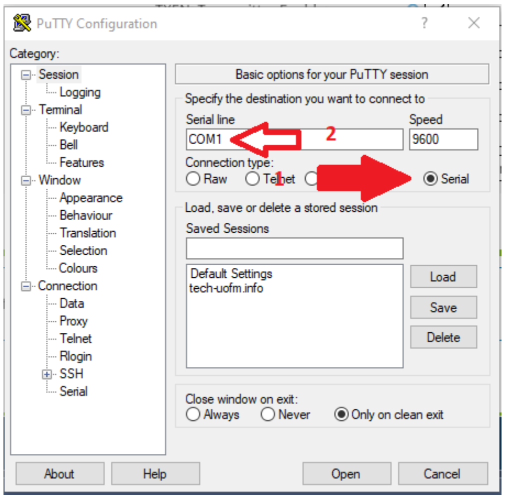
AVR/GNU Linker ➡ Miscellaneous

Other Linker Flags: -lprintf_flt

Save the changes to the linker options (Ctrl + S) and then rebuild the project. Projects that use sprintf and printf type functions should now be able to print floating point numbers to strings or standard output.

To See what you printed using printf

Since no “Screen” is attached to the Arduino, we use the same USB / COM port used to send programs to the board to send messages to a dumb terminal program on the PC. To open this, use the windows search bar to search for “putty” and click on the program found. You should get the following window:



Specify the destination you want to connect to

Serial line: COM1

Speed: 9600

Connection type:
○ Raw ○ Telnet ○ ● Serial

Load, save or delete a stored session

Saved Sessions

Default Settings
tech-uofm.info

Close window on exit:
○ Always ○ Never ● Only on clean exit

Select Serial (1) and then change COM1 to whatever Com port the Arduino is on (same port as the setup for sending code).

NOTE: If you need to change your program and resend it to the Arduino, you will need to CLOSE PUTTY before you send, or you will get an error that says something about the port in use when you try to send the code to the board)