Einrichten des Atmel Studio zum Programmieren eines bootloaders.

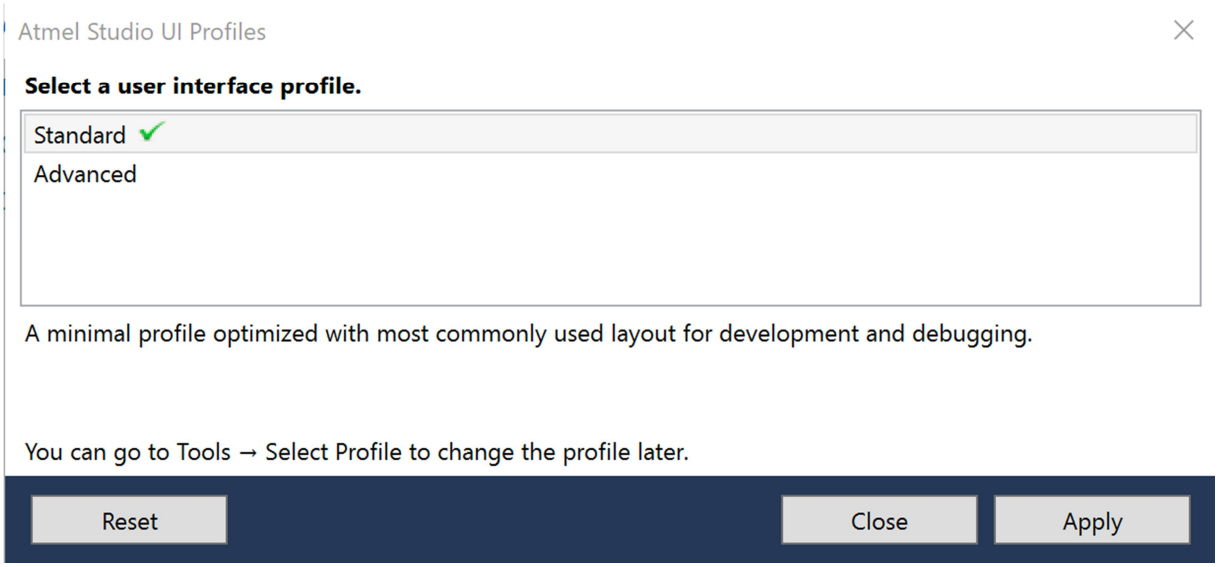
1. Um mit Atmel Studio einen bootloader zu programmieren, wie es z.B. beim Arduino der Fall ist, benötigen wir zuerst die Software "Arduino IDE" denn dort ist das tool "avrdude" integriert das wir einsetzen werden.

   Wir laden das Arduino-Paket als .zip File herunter und entpacken es unter ***C:\tmp***

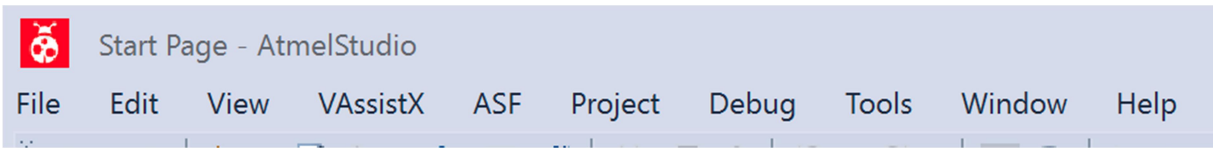
   Den entpackten Ordner benennen wir auf **\Arduino** um.
   Um eine portable Version von Arduino zu generieren erzeugen wir im Arduino-Ordner den Ordner **C:\tmp\Arduino\portable**

2. Jetzt öffnen wir das Programm "Atmel Studio".

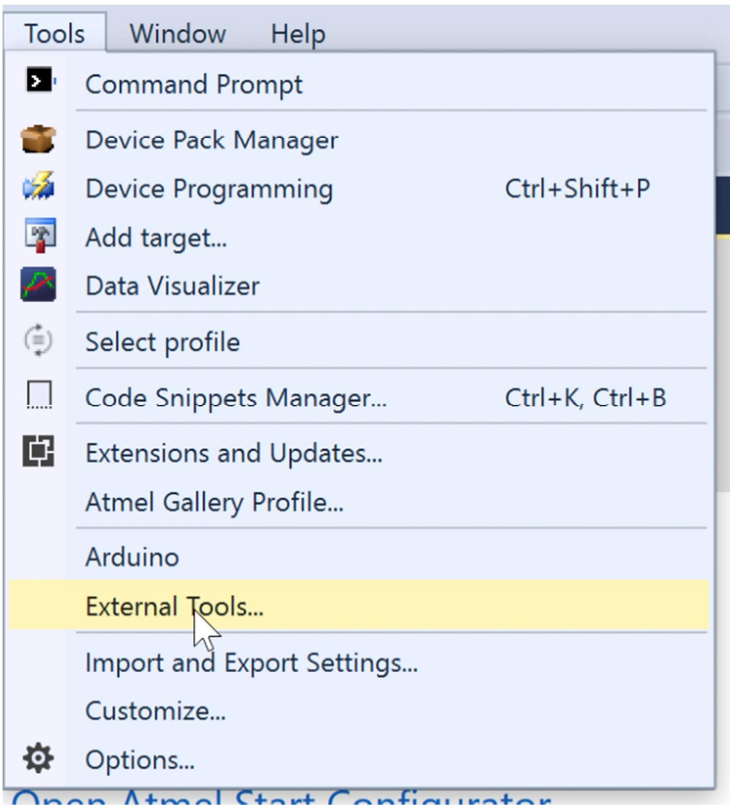
   Bevor wir mit Atmel Studio irgendwas machen müssen wir es noch auf Advanced umstellen, dazu wählen wir oben rechts neben der Orangenen Farbe das Feld "Standard Mode" aus.



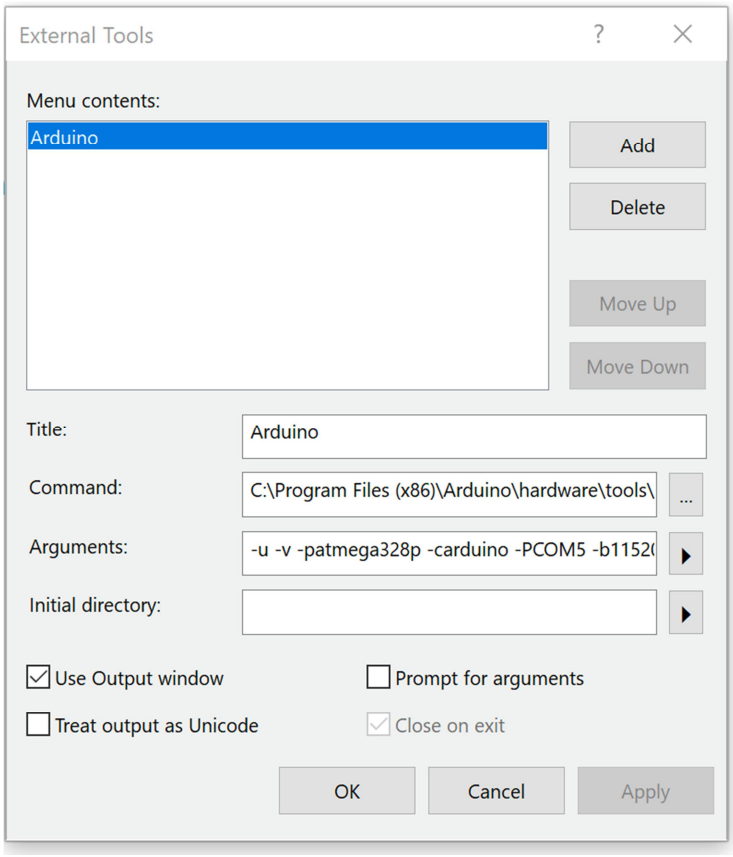
3. Jetzt können wir das tool "avrdude" in Atmel Studio einbinden.
   In Atmel Studio wählen wir in der Toolbar den button "Tools" aus...



   im Bereich "Tools" dann "External Tools..."



Jetzt öffnet sich das Fenster "External Tools..."



Title:
Hier wird der Name des Tool's eingeben. Bei mir heißt das Tool Arduino.

Command:
hier wird der Pfad von unserem "avrdude" angeben.
z.B.: (**C:\tmp\Arduino\hardware\tools\avr\bin**)

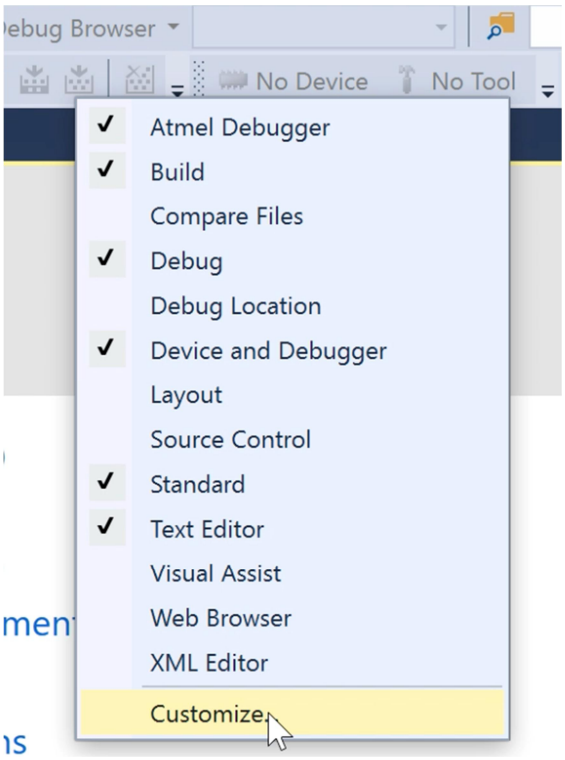
Arguments:
hier geben wir dem tool "avrdude" Anweisungen bei der Übergabe des Hex-Files. Für uns ist nur interessant welchen Com-Port unser Arduino hat. Wenn wir den Port herausgefunden haben müssen wir diesen von

"-PCOM5" in z.B. -PCOM3 ändern.

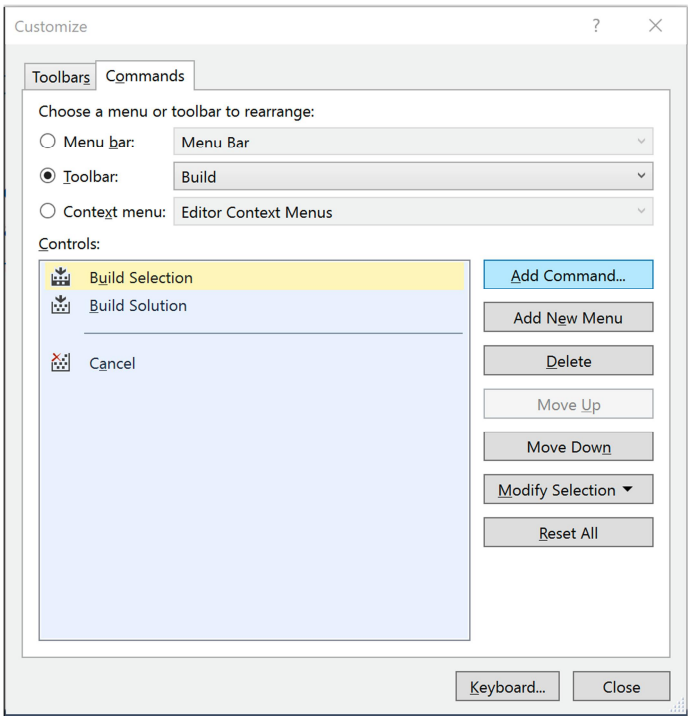
(-u -v -patmega328p -carduino -*PCOM5* -b57600 -Uflash:w:"$(ProjectDir)Debug\$(TargetName).hex":i -C"C:\tmp\Arduino\hardware\tools\avr\etc\avrdude.conf")

Das was in der gelben Klammer steht kopieren und in "Arguments" einfügen.

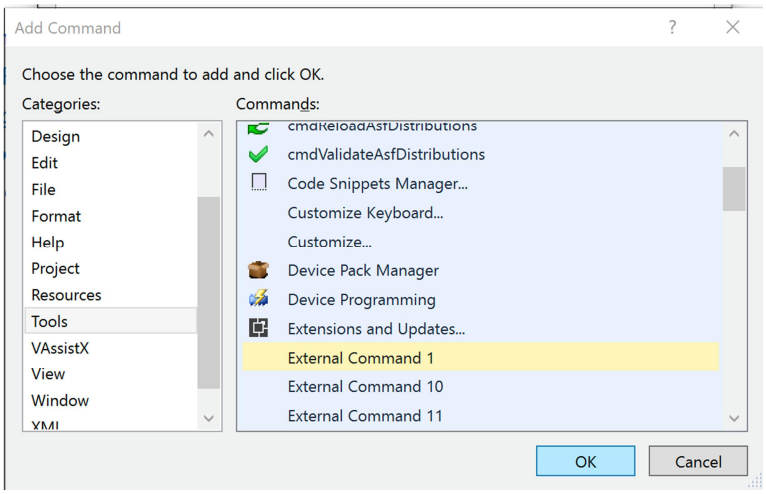
Nach dem wir alle Werte in "External Tools..." angepasst und eingetragen haben müssen wir nur noch die Toolbar anpassen damit wir mit einem Knopf den Arduino programmieren können.



Im Bereich der Toolbar klicken wir mit der rechten Maustaste dann öffnet sich dieses Fenster. Dort klicken wir dann auf ***Customize***.

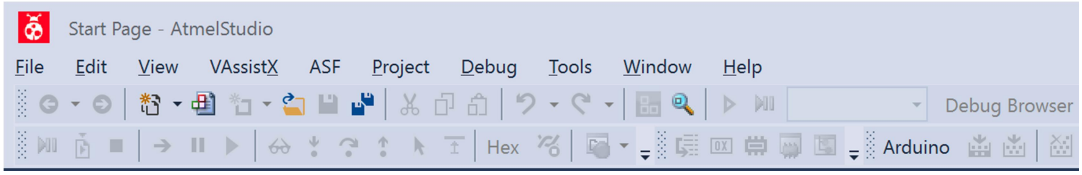


In dem "Customize" Fenster wählen wir "Toolbar:" im Drop down Menu daneben wählen wir "Build" aus. Anschließend klicken wir auf "Add Command..."



Wenn sich das "Add Command" Fenster geöffnet hat wählen wir im linken Fenster unter "Categories" "Tools", und im rechten Fenster bei "Command" "External Command 1" aus und schließen das Fenster mit einem Mausklick auf Ok und das "Customize" Fenster mit Close.

Jetzt sollte neben "Build Solution" Arduino als Tool stehen. Nun können wir mit "Build Solution" den Code übersetzten und anschließend mit dem Button "Arduino" programmieren.



4. Ergänzung: Übergabe der verwendeten Taktfrequenz an die Entwicklungsumgebung