

# Einführung

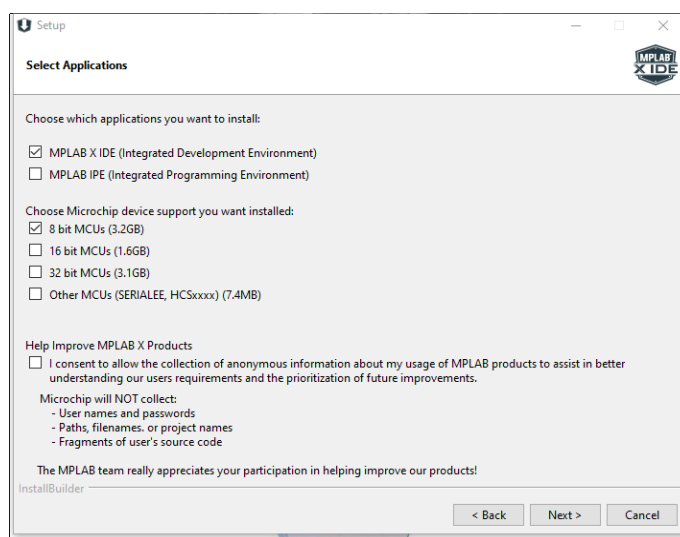
## Projekt Mikrocontroller

Wedel, den 8. Juni 2020

## Installation

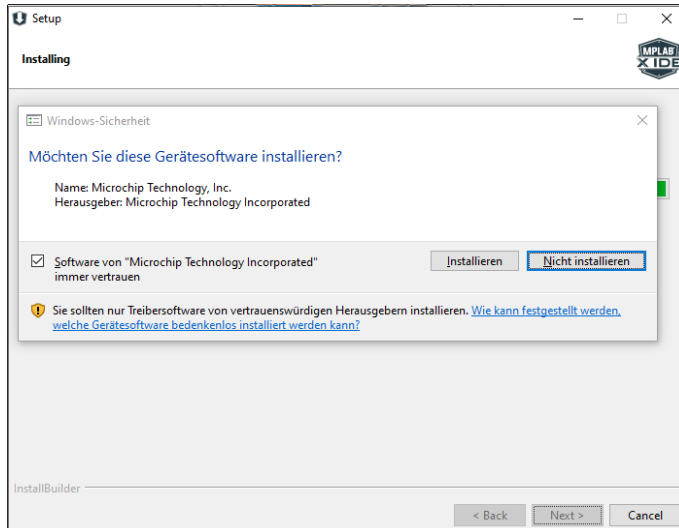
In dieser Veranstaltung wird zum Schreiben des Codes und zum Programmieren der Mikrocontroller die 'MPLAB X IDE' genutzt, die von der Microchip Website heruntergeladen werden kann. Starten Sie den Installations-Wizard, sobald der Download abgeschlossen ist. Klicken Sie auf solange auf 'Next >' bis Sie zum Abschnitt 'Select Applications' kommen.

Sie werden in der Veranstaltung nur mit einem 8 bit Mikrocontroller arbeiten. Daher müssen Sie auch nur die Unterstützungspakete für die 8 bit Controller mit installieren. Das 'MPLAB IPE' werden Sie in dieser Veranstaltung auch nicht benötigen, da die IDE bereits die Programmierung der Controller unterstützt und die IPE lediglich extra Funktionen zur Verfügung stellt.

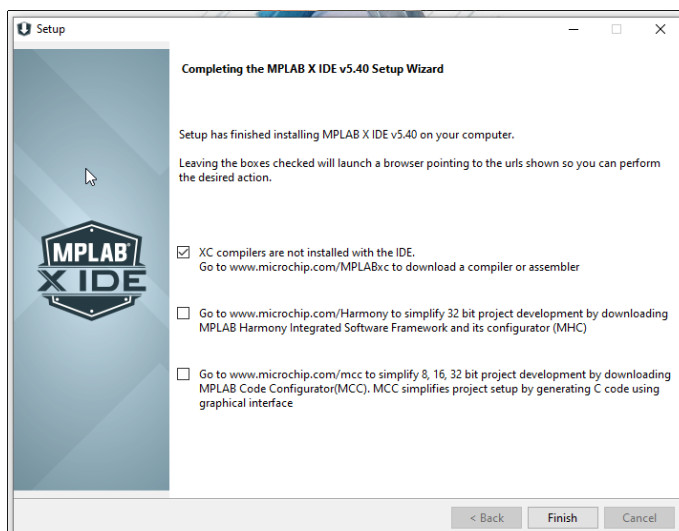


Fahren Sie mit dem Installationsvorgang fort.

Während des Installationsvorgangs werden Sie gefragt ob Sie bestimmte Gerätesoftware installieren wollen. Bitte installieren Sie jede Gerätesoftware die vom Installations-Wizard angefragt wird.

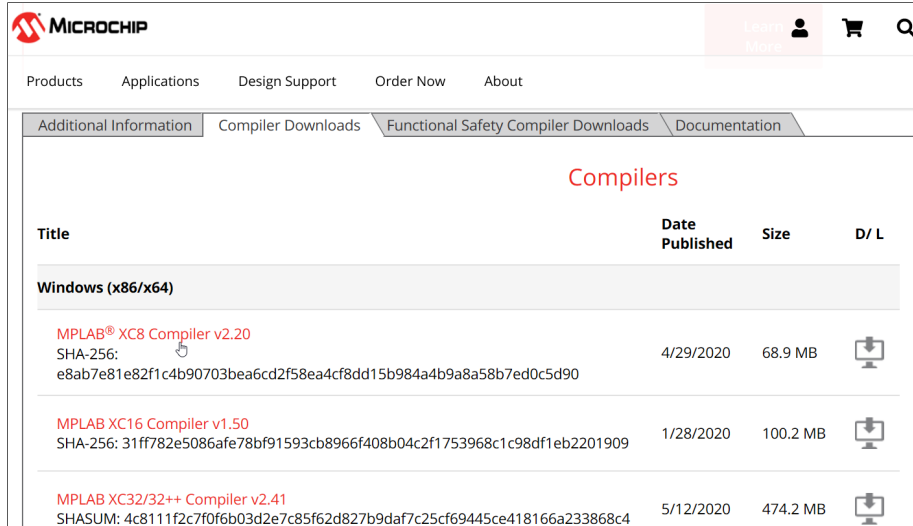





Nachdem der Installationsvorgang erfolgreich abgeschlossen wurde, werden Sie darauf hingewiesen, dass noch kein Compiler installiert wurde.



Wenn Sie das obersten Feld in dem letzten Fenster gesetzt lassen werden Sie, beim Klicken auf 'Finish', zu der Download-Seite des Compilers gebracht.

Wenn die Webseite vollständig geladen ist, scrollen Sie runter bis Sie zu der Karteibox kommen und klicken Sie auf die Kartei 'Compiler Downloads'. Laden sie sich den XC8 Compiler für Ihr Betriebssystem runter.

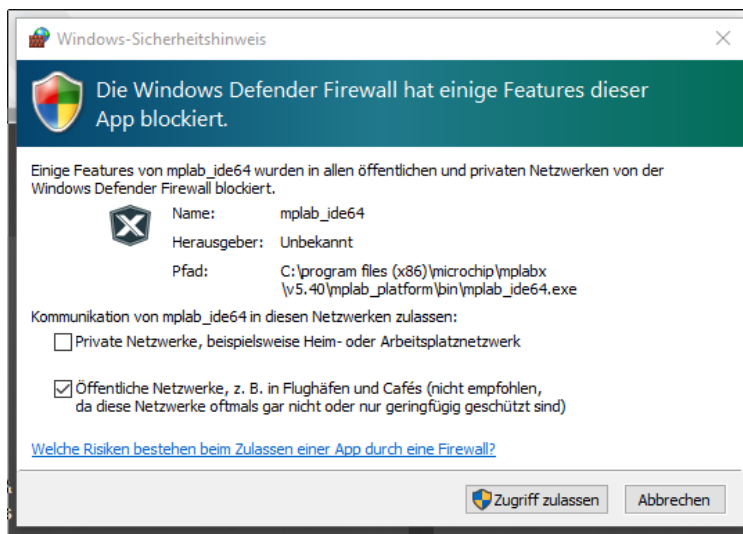


Title	Date Published	Size	D/ L
<b>Windows (x86/x64)</b>			
<b>MPLAB<sup>®</sup> XC8 Compiler v2.20</b> SHA-256: e8ab7e81e82f1c4b90703bea6cd2f58ea4cf8dd15b984a4b9a8a58b7ed0c5d90	4/29/2020	68.9 MB	
<b>MPLAB XC16 Compiler v1.50</b> SHA-256: 31ff782e5086afe78bf91593cb8966f408b04c2f1753968c1c98df1eb2201909	1/28/2020	100.2 MB	
<b>MPLAB XC32/32++ Compiler v2.41</b> SHASUM: 4c8111f2c7f0f6b03d2e7c85f62d827b9daf7c25cf69445ce418166a233868c4	5/12/2020	474.2 MB	

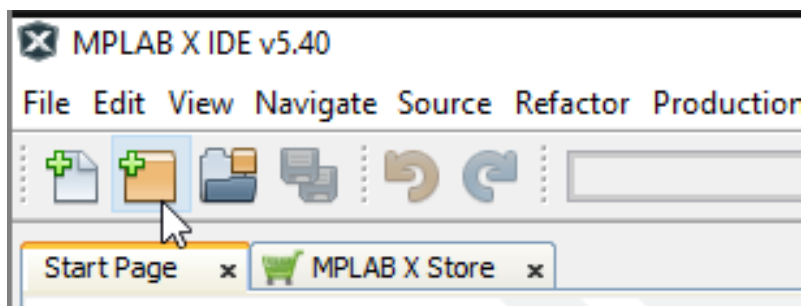
Genauso wie bei der IDE, starten Sie, nach dem herunterladen, die Installation des Compilers. Die Prozedur ist ganz ähnlich. Klicken Sie sich einfach durch den Instalations-Wizard.

## Einrichtung

Nach dem abgeschlossenen Installationsvorgang des Compilers ist die IDE vollständig einsatzfähig. Der nächste Schritt ist die Erstellung eines Test-Projektes um sicher zu stellen, dass alles funktioniert. Bevor sie die IDE zum ersten mal starten, schließen Sie den MPLAB Snap, also den Programmer an. Wenn sie nun unter Windows die IDE starten, wird der Windows Defender sie darauf hinweisen, dass er Features von MPLAB X IDE blockiert hat. Diese Features werden nicht von uns benötigt. Daher können Sie mit 'Abbrechen' das Fenster schließen.

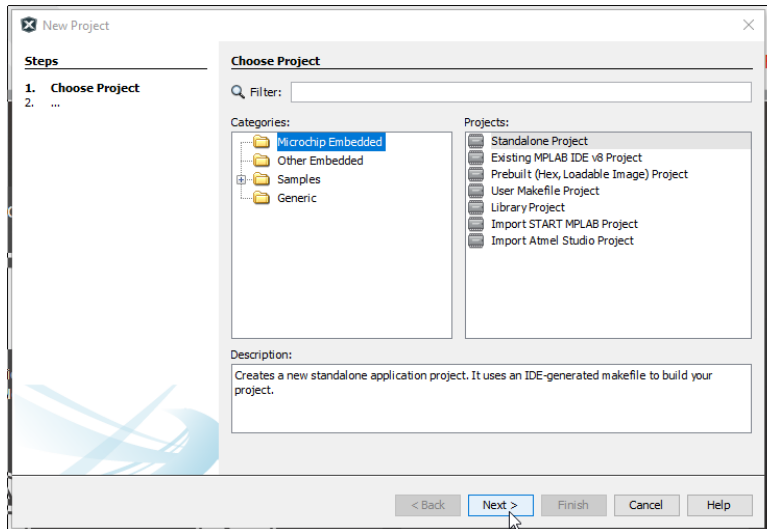


Wenn Sie die IDE starten kommt ihnen eine Oberfläche, ähnlich wie Netbeans entgegen, dass sie bereits aus PS2 kennen sollten. Dem entsprechend sind auch viele Features an den selben Orten zu finden.

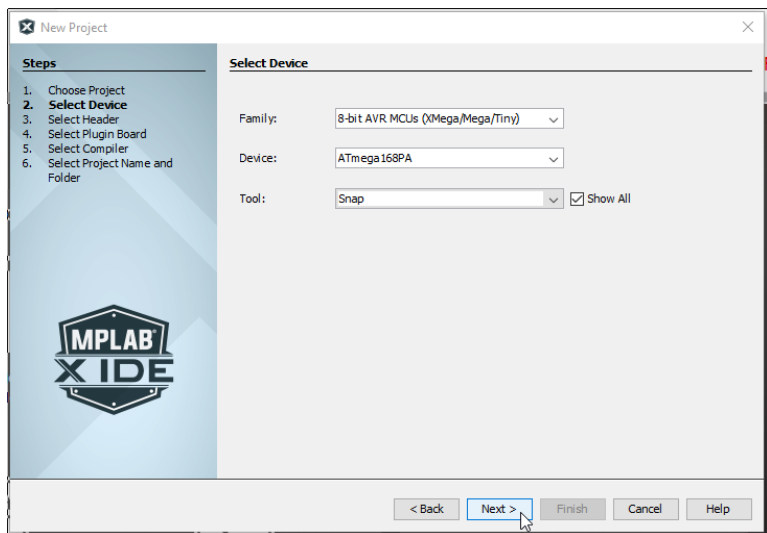


Erstellen sie ein neues Projekt.

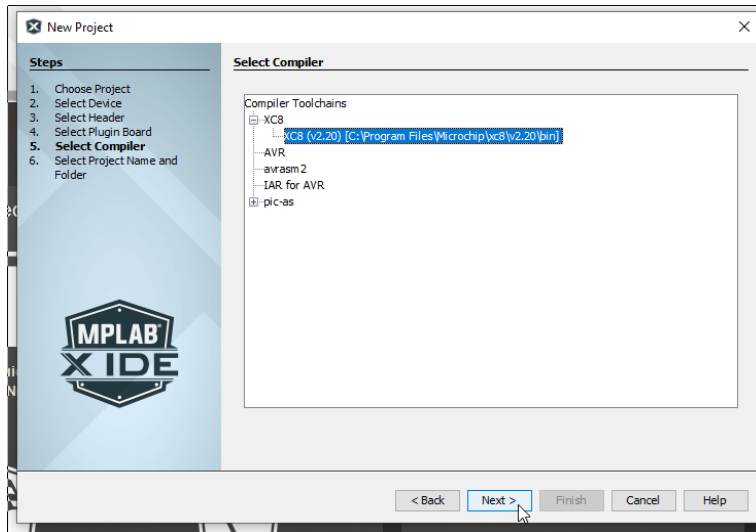
Das Projekt, dass Sie erstellen sollen ist ein 'Microchip Embedded - Standalone Projekt' erstellen. Wählen Sie die entsprechenden Felder unter 'Categories' und 'Projects' aus.



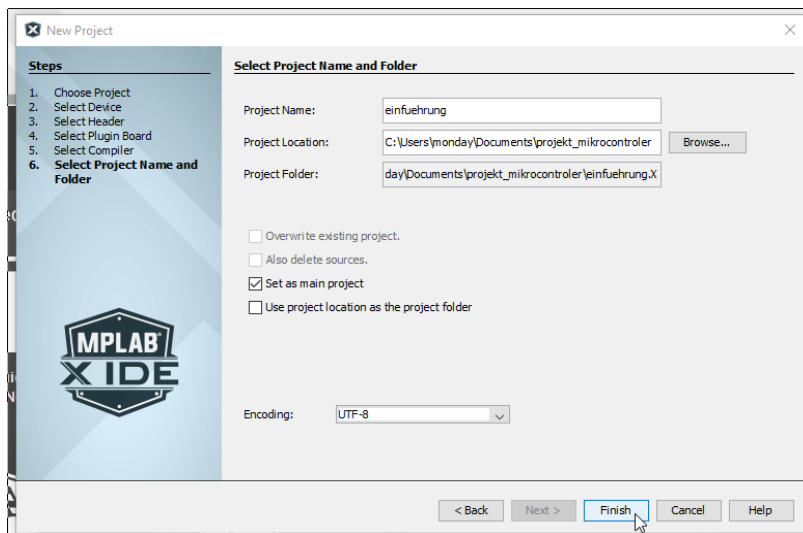
In laufe dieser Veranstaltung werden Sie AVR's ATmega168PA 8 bit Mikrocontroller nutzen. Wählen Sie unter 'Family:' '8-bit AVR MCUs' aus, unter 'Device:' suchen sie den ATmega168PA und unter 'Tool:' wählen sie ihren Programmier, den MPLAB Snap, aus. Dessen Bezeichnung beginnt mit einem 'Snap-SN: BUR' und wird gefolgt von einer Folge von Ziffern.



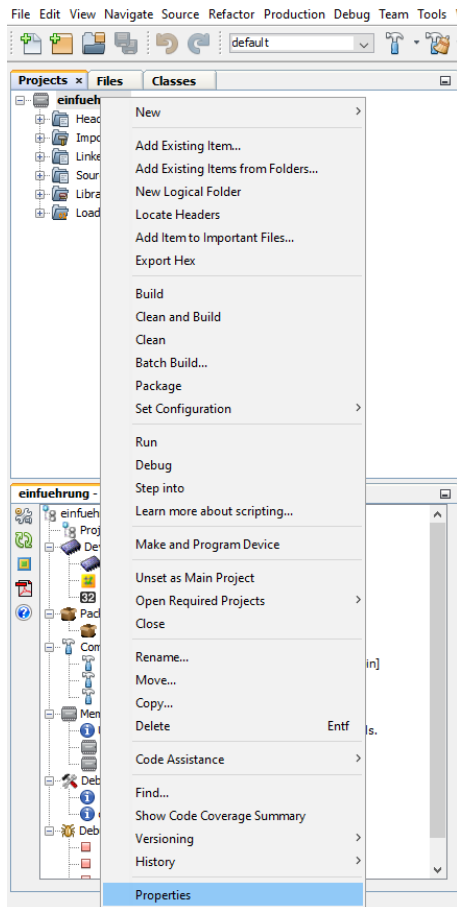
In dem Abschnitt 'Select Compiler' finden Sie unter dem Punkt 'XC8' den von Ihnen installierten Compiler. Wählen Sie diesen und Klicken sie auf 'Next >'.



Geben Sie dem Projekt den Namen 'einfuehrung', wählen Sie einen sinnvollen Ort an dem Sie das Projekt auch wiederfinden und wählen sie als Encoding für die Source und Header Files 'UTF-8' aus. Nun können sie mit dem Klicken auf 'Finish' die Projekterstellung beenden.

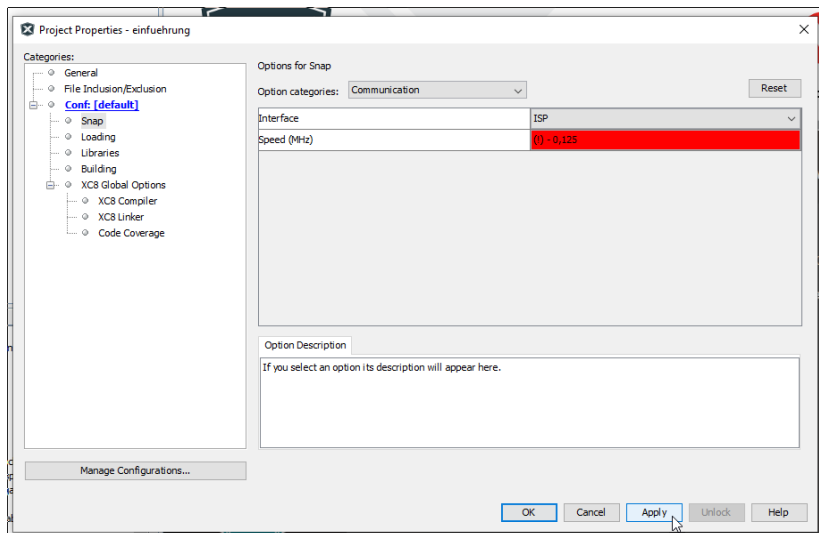


Für weitere Konfiguration des Projektes gehen Klicken Sie im Kontext-Menü des Projektes auf 'Properties'.

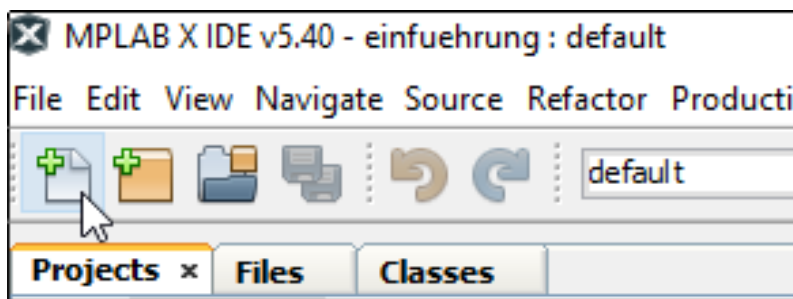




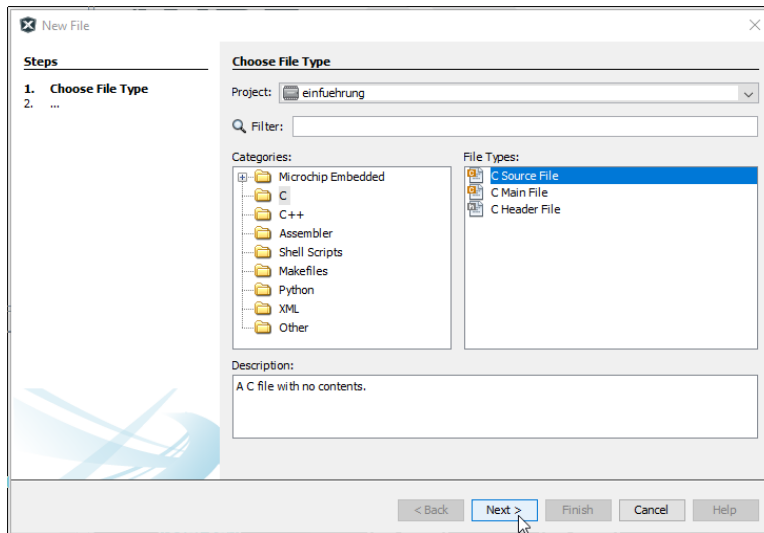
In Project Properties klicken sie auf den Unterpunkt 'Snap'. Wählen Sie unter 'Option categories' 'Communication' aus. Als Interface wählen Sie 'ISP' und unter 'Speed(MHz)' schreiben sie den Wert 0,125. Es wird Ihnen als Fehler angezeigt werden, doch dies können sie ignorieren. Bestätigen sie ihre Anpassungen mit 'Apply' und schließen sie das Fenster mit 'OK'.



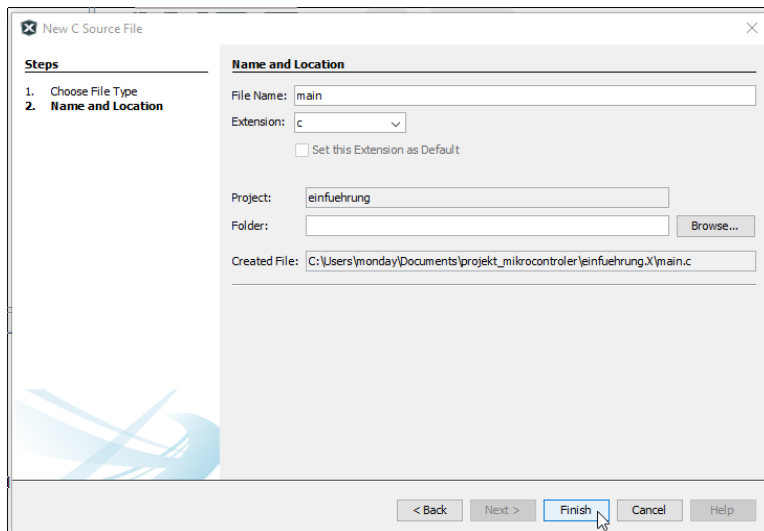
Nun müssen sie noch ein eine Datei erstellen, in der sie Ihren Code schreiben.



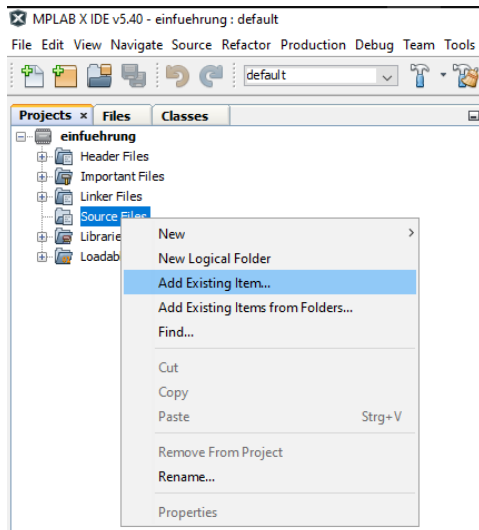
Erstellen sie eine Main-File aus.



Nennen sie die Datei 'main' und klicken sie auf 'Finish'



Wenn die Datei nicht sofort in dem Projekt, auf der linken Seite auftaucht, müssen Sie diese selbst hinzufügen. Dafür rechts-klicken Sie auf das Ordner-Symbol 'Source' und klicken auf 'Add Existing Item...'.  
.



Wählen Sie die main.c Datei aus und drücken sie Enter. Nun sind Sie bereit anzufangen.

Beginnen sie damit eine LED zum Blinken zu bringen und eine Call&Responnds über UART zu implementieren. Dies dient dem Zweck zu überprüfen, ob Hardware und Software einwandfrei funktionieren. Wenn sie Material brauchen oder Fragen haben, melden sie sich VORHER bei Timm Bostelmann oder A. Sokol (iat103313@fh-wedel.de) an und machen sie einen Termin aus.