

Aufgabe: Schreiben der iButton - Chipseriennummer über WLAN in eine SQL-Datenbank**Die Hardware Komponenten sind:**

ARDUINO ESP32; Touch-Display, iButton Microleser und Leiterplatte.

Der Touchscreen wurde gewählt, um das WLAN-Kennwort einzugeben. Selbstverständlich dient er noch, um Meldungen anzuzeigen. Der Vorteil des iButton ist, dass dieser als Anschlussleitung nur eine Masseleitung und eine Datenleitung benötigt. Diese dürfen je nach Qualität bis zu 15m lang sein. Wie der iButton funktioniert wird als bekannt vorausgesetzt.

Der Schwerpunkt dieser Anleitung liegt nicht etwa darin, eine fertige Applikation in ein ARDUINO-System zu installieren. Sie soll eine Möglichkeit für eigene Kreativität bieten und zeitaufwändige Vorarbeiten reduzieren.

Entwicklungssystem für ARDUINO:

Die Anleitung richtet sich an Menschen, die C++ für ARDUINO programmieren wollen. Die Grundlage hierfür ist ein Entwicklungssystem. Ein besonders wichtiges Merkmal eines Entwicklungssystems ist, ein Testprogramm schrittweise auszuführen um die Zustände beobachten zu können. Unersetzlich ist die Möglichkeit sog. Breakpoints im Programm zu setzen die das Programm an einer definierten Stelle anhalten, um je nach Situation die Zustände zu kontrollieren

Softwarekomponenten

Das Entwicklungssystem setzt auf einem „jungfräulichen“ Windows 10 (aktuell 1809) auf. Die Entwicklungsoberfläche ist das kostenlose Microsoft Visual Studio. Darauf wird als sog. Plugin die leider kostenpflichtige Entwicklungsumgebung Visual Micro installiert, die auf der unumgänglich notwendigen Original ARDUINO IDE aufsetzt. Diese liefert im Wesentlichen die passenden Hardwaretreiber.

Die nachfolgenden Screenshots leiten durch den Dschungel der Optionen. Der Anwender erhält so die minimal erforderlichen Komponenten um seine Sketche direkt auf die ESP32 Hardware zu laden. Stolperfallen sind das Holen und Einbinden der erforderlichen Bibliotheken, sowie die Anpassung der Hardware bzw. Anpassung der Hardware-Dateien

Teil 1: Installation MS Visual Studio + Visual Micro+ARDUINO IDE

Teil 2: Installation und Aktivierung des Boards

Teil 3: Upload bestehendes Sketch nach ESP32 und Debugging

Teil 4: Hardwareaufbau ESP32 Anpassung der Verdrahtung an die Bibliotheken

Teil 5: Kontakt mit WLAN

Teil 6: Datenübermittlung der TSN in eine mysql-DB

Die verwendeten Abkürzungen:

VS: Visual Studio. Die Microsoft Entwicklungsumgebung

VS Micro: Das ARDUINO-Plugin für Visual Studio

MS: Microsoft

ARDUINO IDE: Die Original ARDUINO Entwicklungsumgebung. Minimale Möglichkeit um Programme zu schreiben und auf die Hardware zu laden.

Teil 1.1 : Installation MS Visual Studio

Ausgangspunkt: rohes Betriebssystem (Win 10 x64 1809)

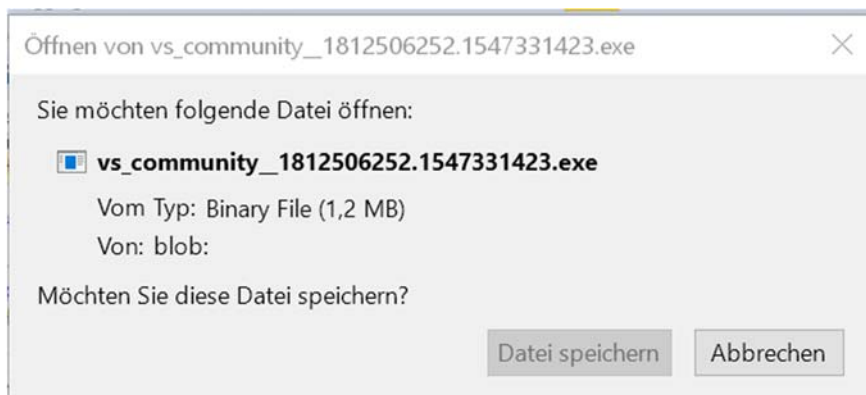
Benötigt werden: VS (kostenfrei) VM (25€) Visual Micro; Arduino IDE (kostenfrei)

URL zu MS Visual Studio:

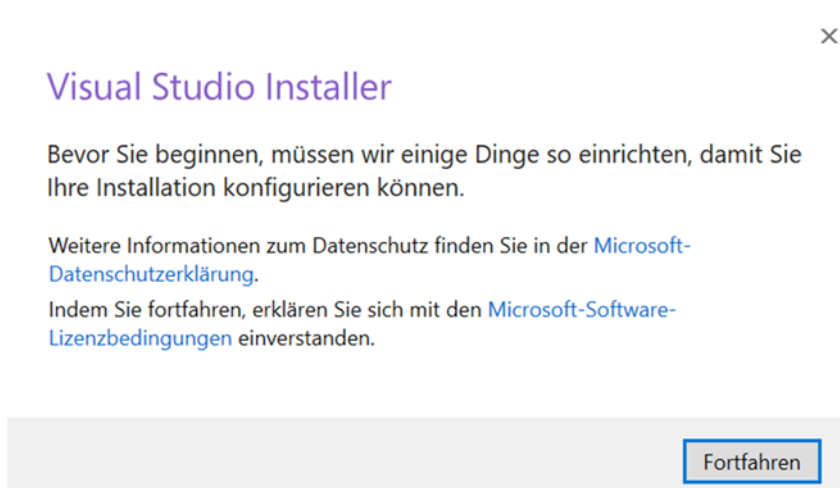
<https://visualstudio.microsoft.com/de/downloads/?rr=https%3A%2F%2Fwww.microsoft.com%2Fde-de%2Fsearch%3Fq%3Dvisual%2Bstudio%2Bdownload>

URL zu MS Visual Studio download kostenfreie Version: (empfohlen)

<https://visualstudio.microsoft.com/de/thank-you-downloading-visual-studio/?sku=Community&rel=16>



Start Visual Studio 2019 Installer :

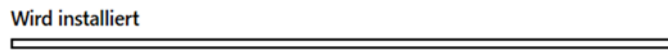


„Fortfahren“

Visual Studio Installer

Einen Moment bitte... Ihre Dateien werden abgerufen.

Wird heruntergeladen: 8,25 MB von 63,82 MB 452,17 KB/s



Abbrechen

Wird geändert – Visual Studio Community 2019 – 16.0.3

Workloads

Einzelne Komponenten

Sprachpakete

Installationspfade

von Node.js, einer asynchronen, ereignisgesteuerten...

Windows (3)



.NET-Desktopentwicklung



Erstellen von WPF-, Windows Forms- und Konsolenanwendungen mithilfe von C#, Visual Basic und...



Desktopentwicklung mit C++



Erstellen Sie Windows-Desktopanwendungen mit dem Microsoft C++-Toolset, ATL oder MFC.



Entwicklung für die universelle Windows-Plattform




Hiermit erstellen Sie Anwendungen für die universelle Windows-Plattform mit C#, VB oder optional C++.

Nur C++ wählen, und „Installieren“ klicken

Visual Studio Installer

Installiert Verfügbar

 **Visual Studio Community 2019** Anhalten
Download und Überprüfung werden durchgeführt: 5 MB von 1,65 GB (581 KB/Sek.)
0 %
Wird installiert: Paket 8 von 329
0 %
Microsoft.VisualStudio.NativelImageSupport
 Nach der Installation starten
[Anmerkungen zu dieser Version](#)

Für bereits bestehende Visual Studio Installer Versionen

Wird auf „Aktualisieren“ geklickt

Visual Studio Installer


Installiert Verfügbar

 **Visual Studio Community 2019** Aktualisieren
16.0.1 Starten
 Update verfügbar Mehr ▾
16.0.3

„Aktualisieren“

Visual Studio Installer

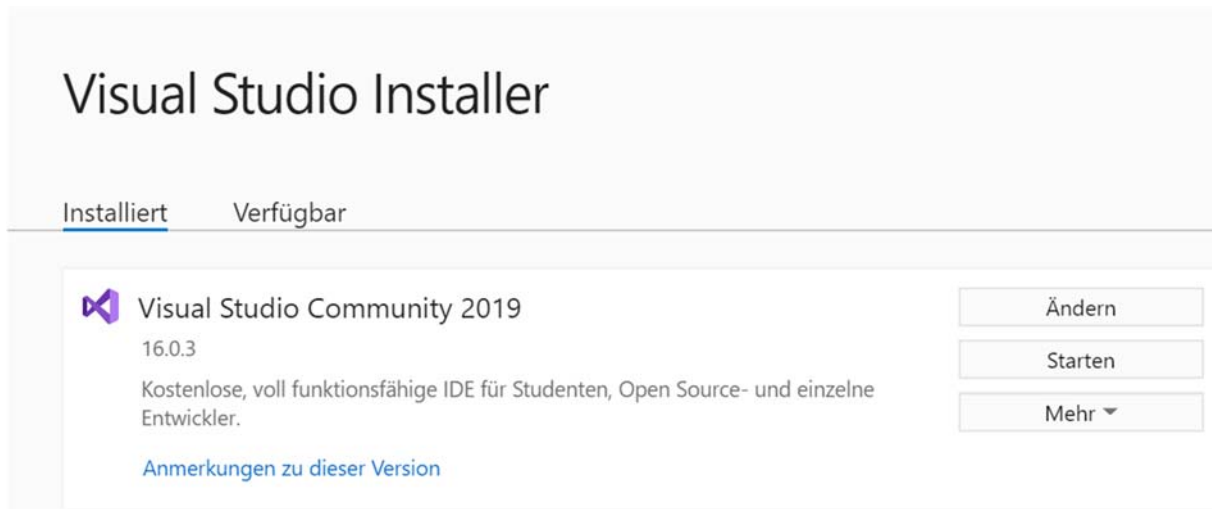
Installiert Verfügbar

 **Visual Studio Community 2019** Ändern
16.0.3 Starten
Kostenlose, voll funktionsfähige IDE für Studenten, Open Source- und einzelne Entwickler. Mehr ▾
[Anmerkungen zu dieser Version](#)

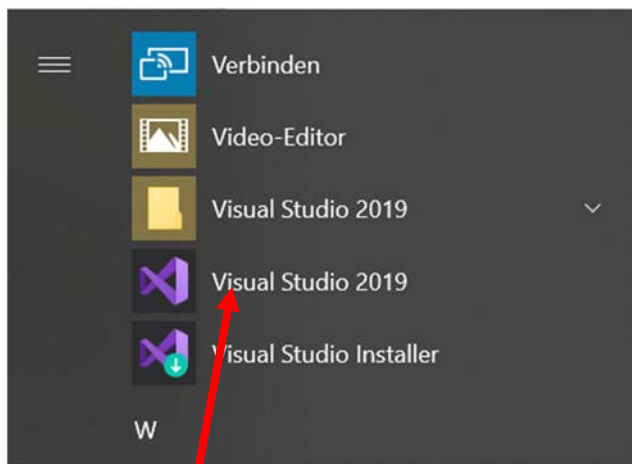
„Ändern“

Dann kann der Installer beendet werden. („Schließen“), damit ist der Installationsvorgang beendet

Teil 1.2 : Programmstart MS Visual Studio

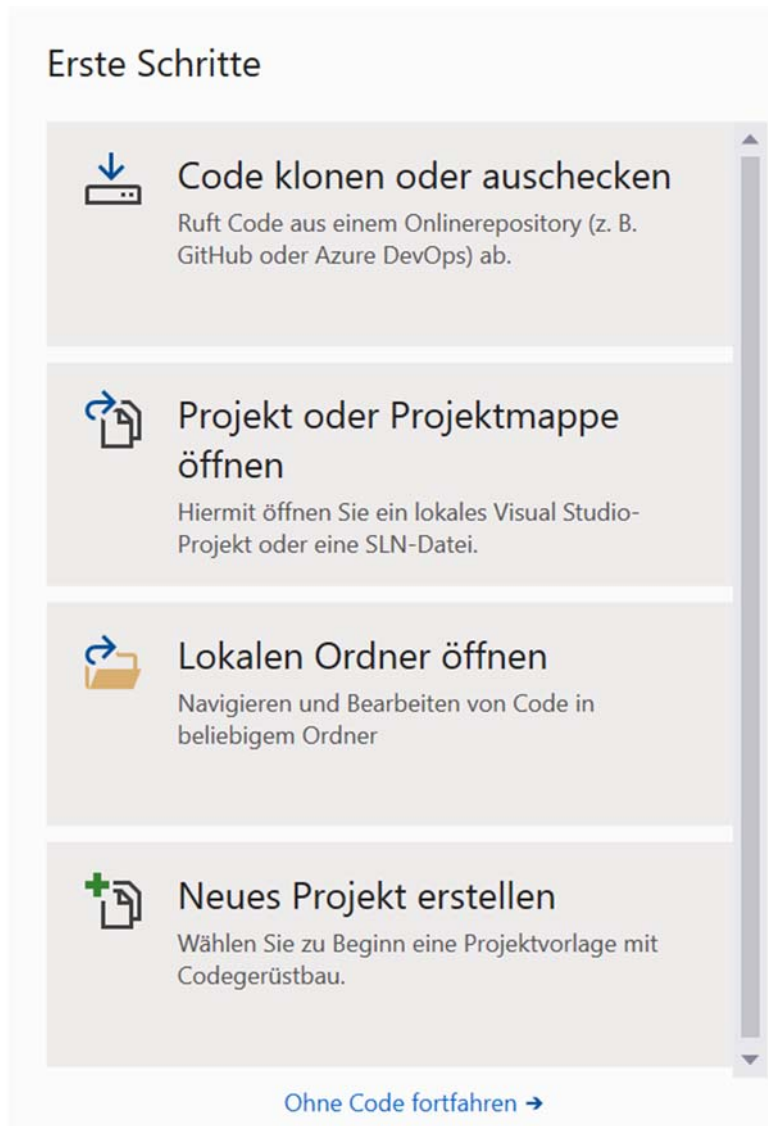


Zum Programmaufruf des VS entweder hier „Starten“ klicken, oder über das Startmenü gehen:



Hier klicken:

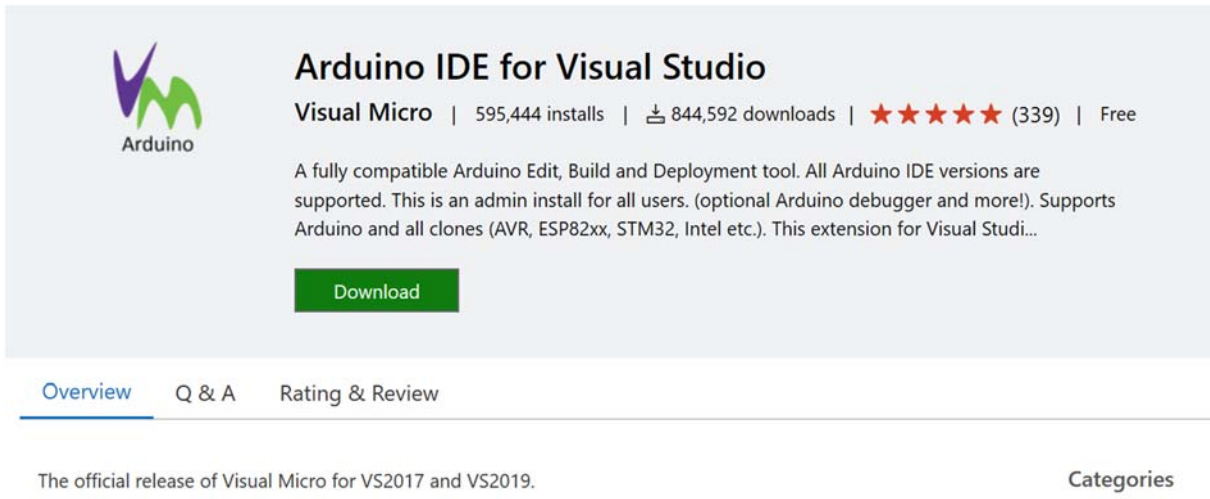
Teil 1.3 : Der erste Start von VS: ein leeres Projekt aufrufen



Bei Klick auf „Ohne Code fortfahren“ öffnet sich ein leeres VS-Fenster mit den default Einstellungen für C++
Bei Klick auf „Neues Projekt erstellen“ öffnet sich ein Wizzard. Weiter mit den default Einstellungen erzeugt ein leeres Projekt „Projekt1“. Dies Einzelheiten hierzu werden später behandelt. Die jetzt folgenden Schritte behandeln das Vorgehen um die ARDUINO IDE in VS einzubinden.

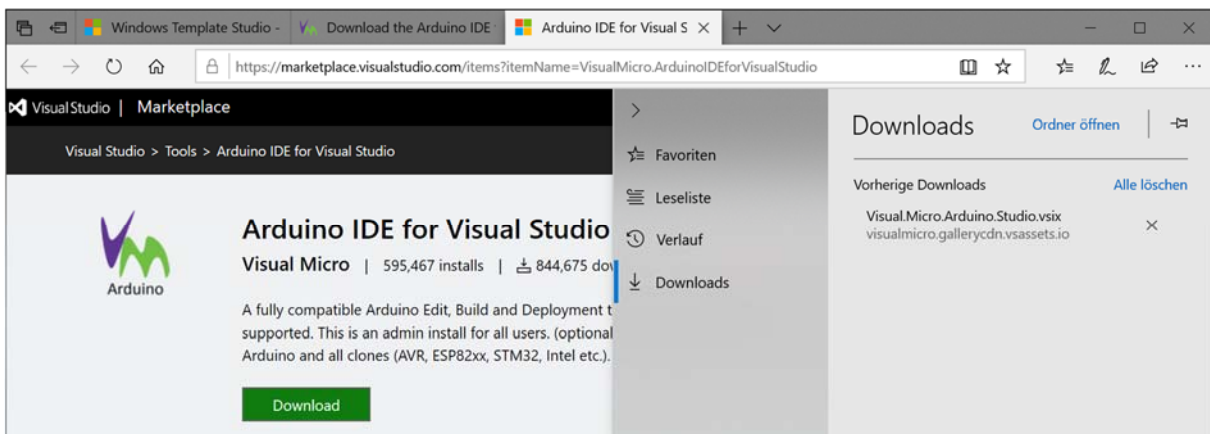
Mit folgendem URL-Aufruf gelangt man in das Visual-Micro Downloadportal:

<https://marketplace.visualstudio.com/items?itemName=VisualMicro.ArduinoIDEforVisualStudio>



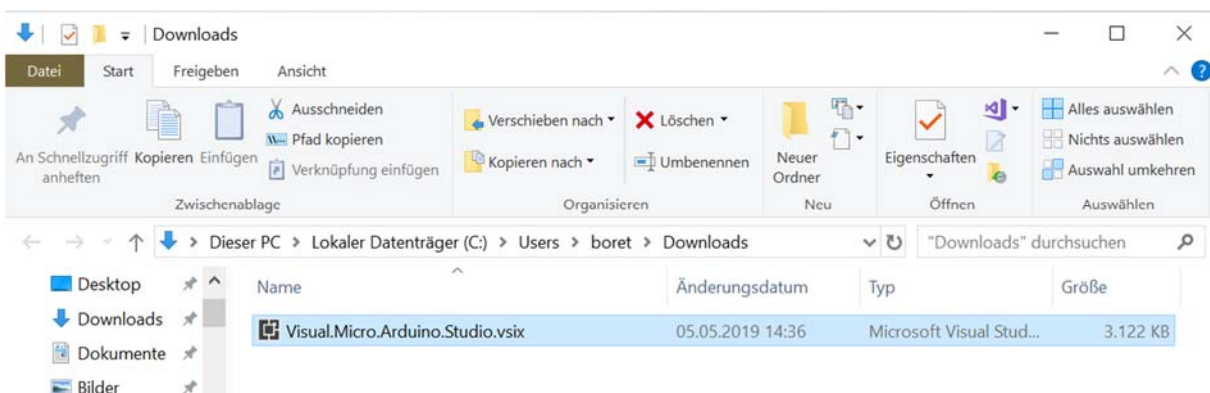
The screenshot shows the Visual Studio Marketplace page for the 'Arduino IDE for Visual Studio' extension. The page features the Arduino logo, the extension name, and statistics: 595,444 installs, 844,592 downloads, and a 5-star rating from 339 reviews. A green 'Download' button is prominent. Below the main content, there are tabs for 'Overview', 'Q & A', and 'Rating & Review'. A description states it is a fully compatible tool for all Arduino versions and clones. At the bottom, it notes it is the official release for VS2017 and VS2019.

„Download“

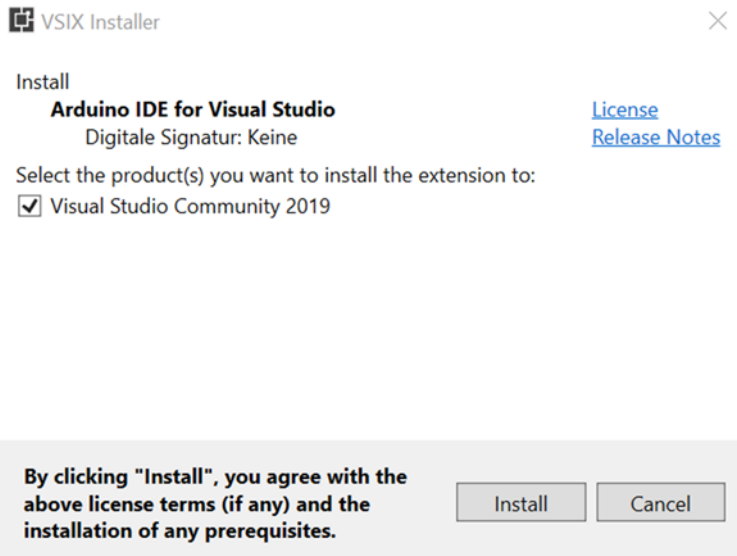


This screenshot shows a web browser window with the Visual Studio Marketplace page. The 'Downloads' panel on the right side of the browser is open, showing a list of 'Vorherige Downloads' (Previous Downloads). The file 'Visual.Micro.Arduino.Studio.vsix' is listed with its source 'visualmicro.gallerycdn.vsassets.io'. The 'Downloads' button on the marketplace page is highlighted.

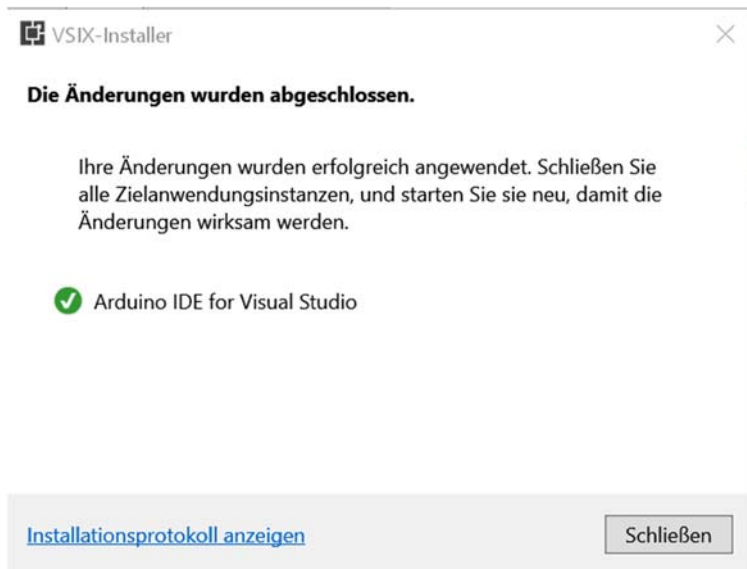
Visual.Micro.Arduino.Studio.vsix downloaden und zum Installieren klicken:



The screenshot shows a Windows File Explorer window open to the 'Downloads' folder. The file 'Visual.Micro.Arduino.Studio.vsix' is selected. The file's properties are visible: Name: Visual.Micro.Arduino.Studio.vsix, Änderungsdatum: 05.05.2019 14:36, Typ: Microsoft Visual Stud..., Größe: 3.122 KB. The Windows taskbar and navigation pane are also visible.



„Install“ klicken



Allgemeines zur WINDOWS Speichernutzung:

Folgende zwei Dateisystem-Bereiche sind für Visual Micro vorgeschlagen:

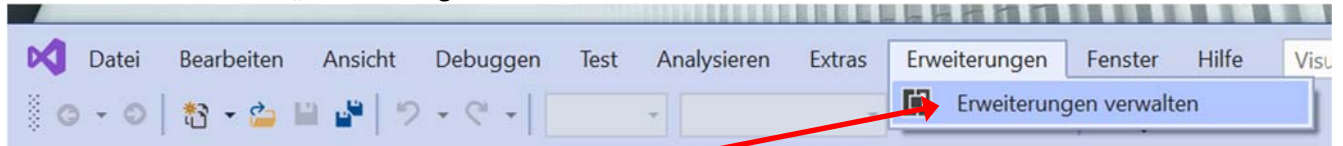
Current IDE-Folder: C:\Program Files (x86)\Arduino : Programme Arduino und Standard Daten

User Hardware Folder: C:\Users*benutzername*\Documents\Arduino\hardware

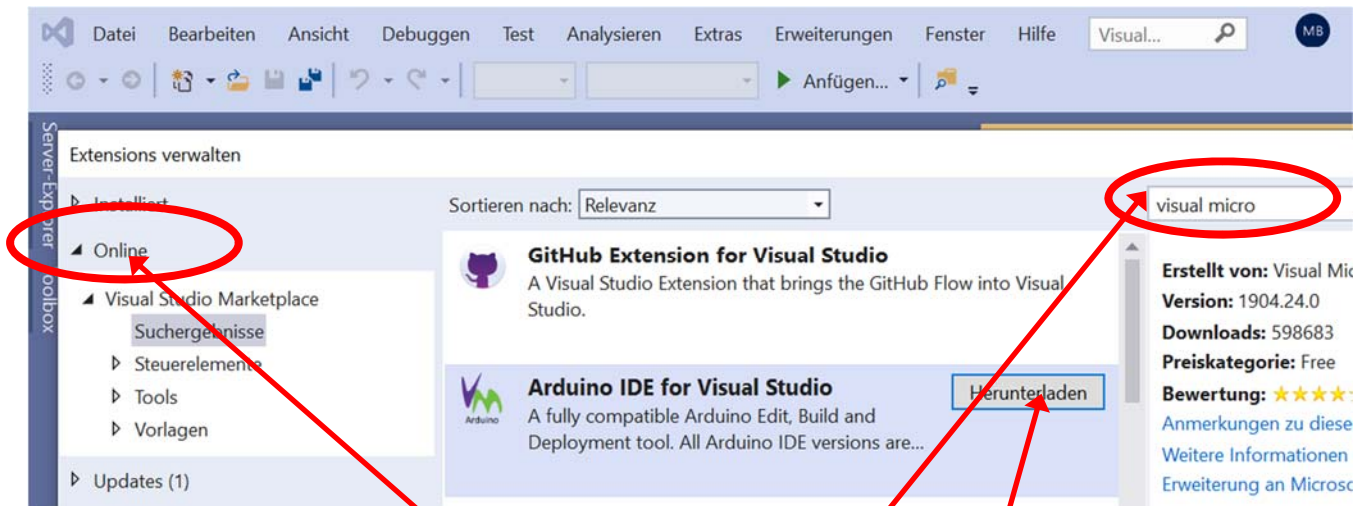
der Benutzer speichert hier seine „Werke“

Auf diese Speicherbereiche sind VS Micro und ARDUINO IDE von Haus aus voreingestellt. Am besten übernimmt man diese Voreinstellung.

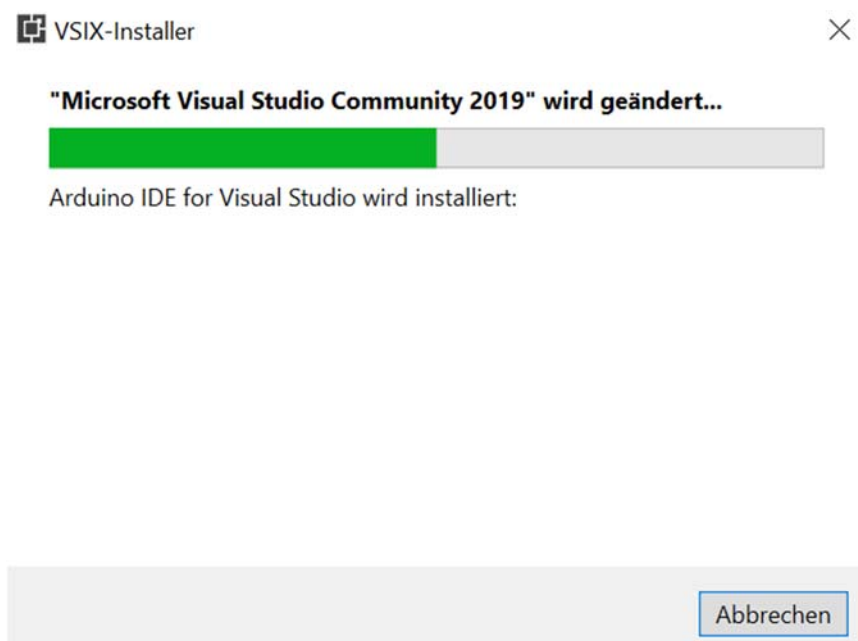
Über die VS- Menüzeile „Erweiterungen“ lokalisieren



„Erweiterungen verwalten“ klicken




Im sich darauf öffnenden Fenster „online“ wählen, in die Suchzeile „visual micro“ eintragen. Danach bekommt man „Arduino für Visual Studio“ angeboten. Dies anklicken und „Herunterladen“ klicken und den Anweisungen folgen. Es startet der VS-Installer:



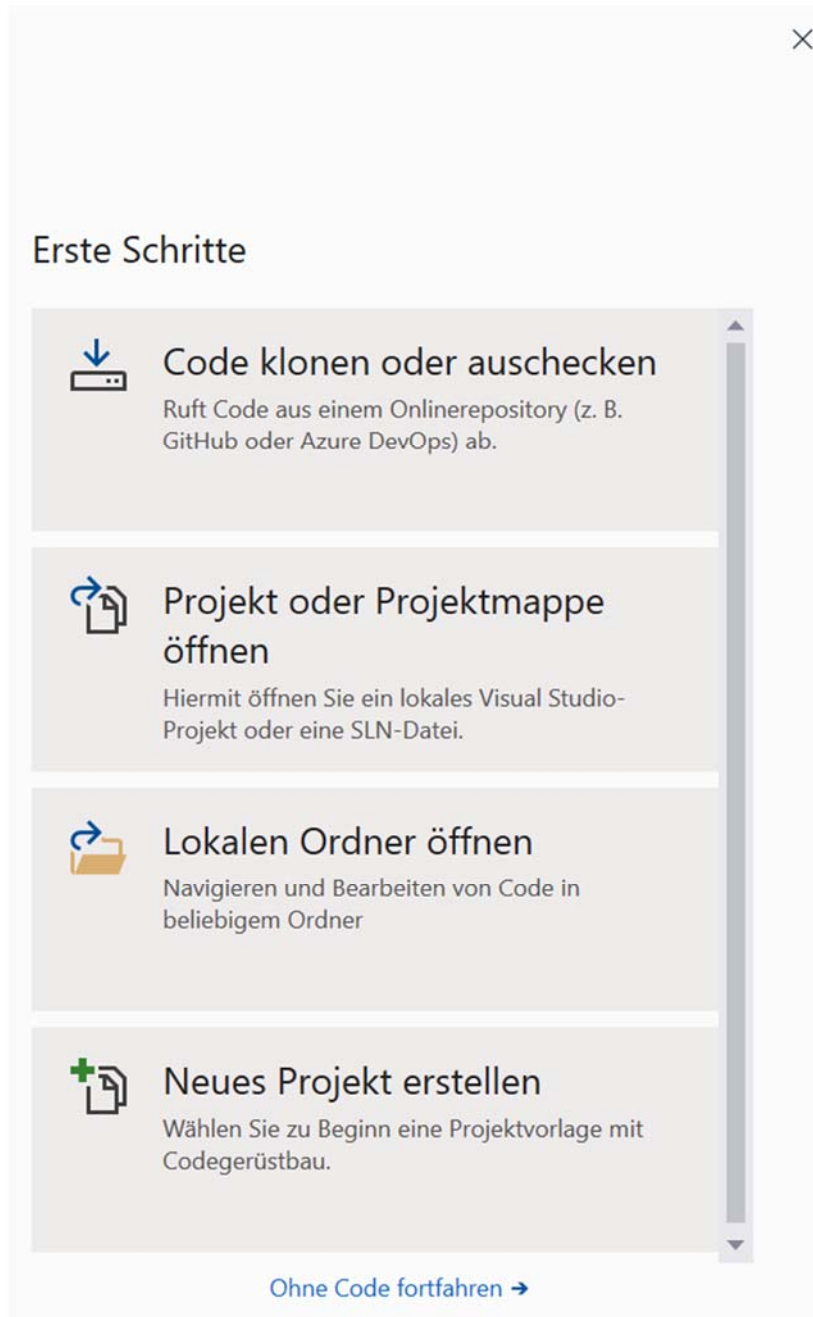
VSIX-Installer ✕**Die Änderungen wurden abgeschlossen.**

Ihre Änderungen wurden erfolgreich angewendet. Schließen Sie alle Zielanwendungsinstanzen, und starten Sie sie neu, damit die Änderungen wirksam werden.

 Arduino IDE for Visual Studio

[Installationsprotokoll anzeigen](#)[Schließen](#)

Neustart VS:



„Ohne Code fortfahren“

Direkt nach VS Neustart will Visual Micro die Location der Arduino Entwicklungsumgebung kennen:

Configure Ide Locations

Please specify a micro-controller Ide location

Visual Micro needs to know where, on your computer, application(s) such as the Arduino.exe are located.

If an application is not already installed then please download it using the 'Download/Install' button.
Support for some platforms is still under development and we value feed back in our forum

Arduino 1.6/ Use installed IDE. Provides built-in hardware, reference/help and libraries.

* Enter the ide folder location (example: c:\arduino)

Optional sketchbook location (best to leave empty, also affects the location of libraries/hardware)

Optional additional boards manager urls (url1,url2,url3 ...). Warning: Use safe urls from the link below

<https://github.com/arduino/Arduino/wiki/Unofficial-list-of-3rd-party-boards-support-urls>

Download/Install Ide OK Cancel

Das Verz. **C:\Program Files\Arduino** händisch anlegen (Adminrechte). Wie das geht wird als bekannt vorausgesetzt.

Hilfestellung für die korrekte Schreibweise:


Mit dem Explorer in das Standard-Verzeichnis C:Programmfiles\X86 navigieren:



Diesen String in die Zwischenablage kopieren und als Location eingeben C:\Program Files (x86)\Arduino (würde sich aus der ARDUINO IDE holen, die jetzt nicht da ist am besten jetzt auf „OK“ klicken).

Dies sieht dann so aus:

Configure IDE Locations ✕

 **Please specify a micro-controller IDE location**

Visual Micro needs to know where, on your computer, application(s) such as the Arduino.exe are located.

If an application is not already installed then please download it using the 'Download/Install' button.
Support for some platforms is still under development and we value feed back in our forum

Arduino 1.6/ Use installed IDE. Provides built-in hardware, reference/help and libraries.

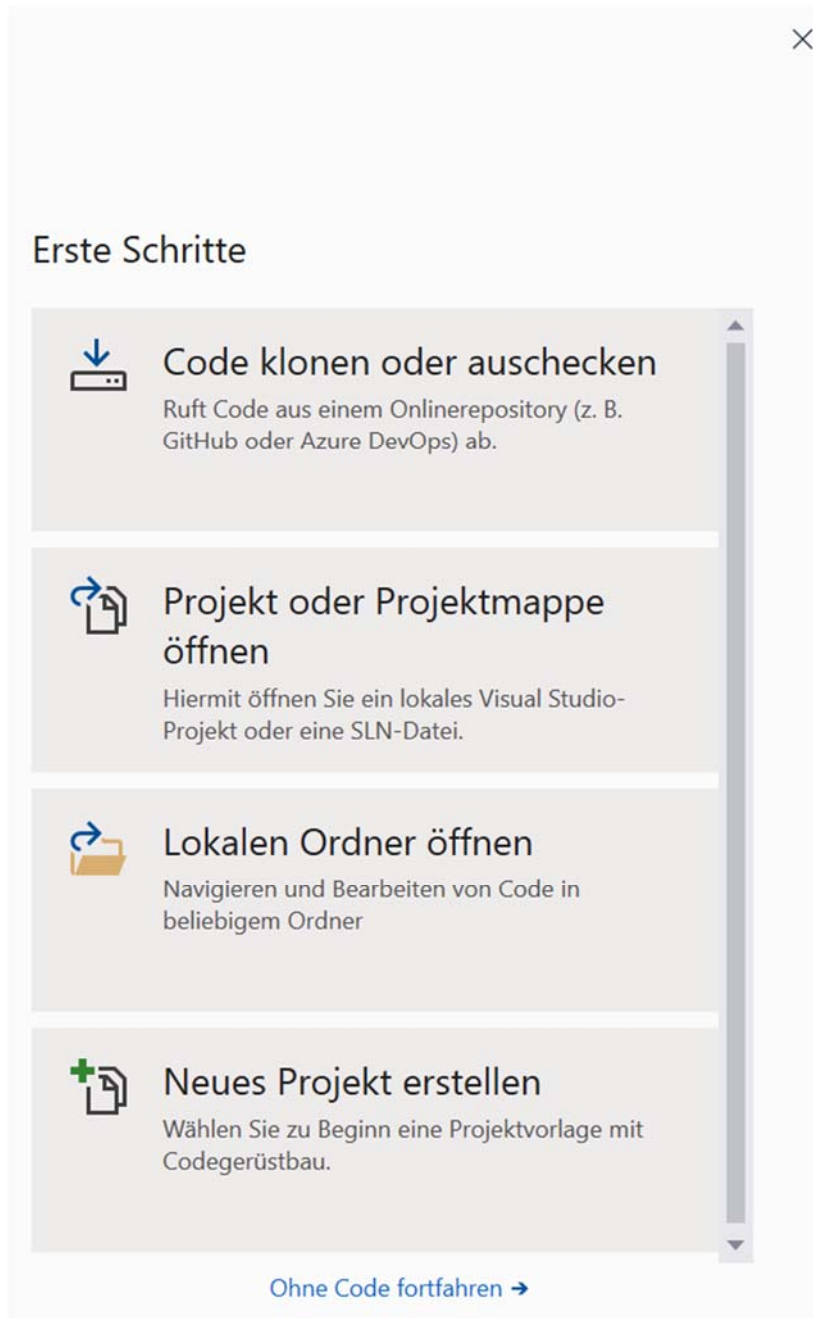
*** Enter the ide folder location (example: c:\arduino)**

Optional sketchbook location (best to leave empty, also affects the location of libraries/hardware)

Optional additional boards manager urls (url1,url2,url3 ...). Warning: Use safe urls from the link below

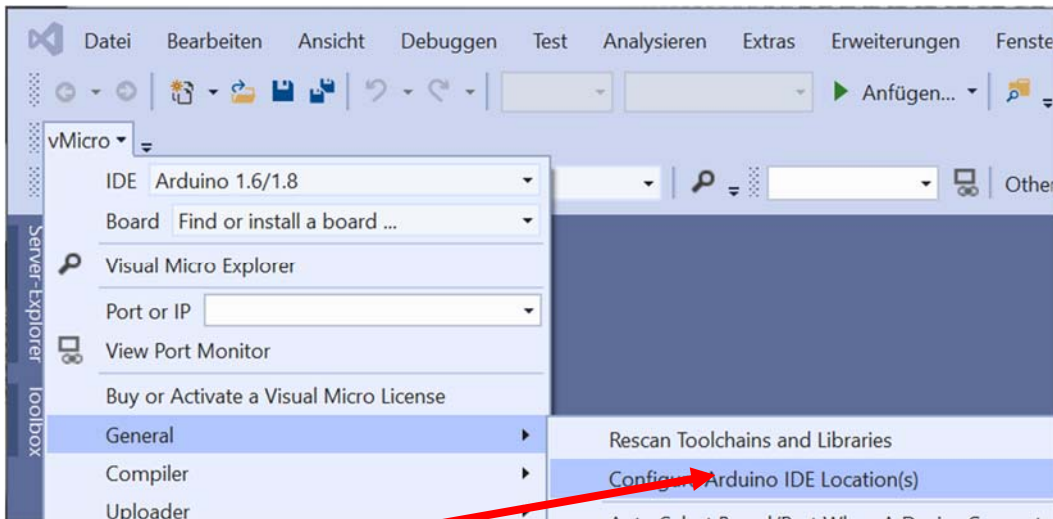
<https://github.com/arduino/Arduino/wiki/Unofficial-list-of-3rd-party-boards-support-urls>

„OK“ Klicken danach:

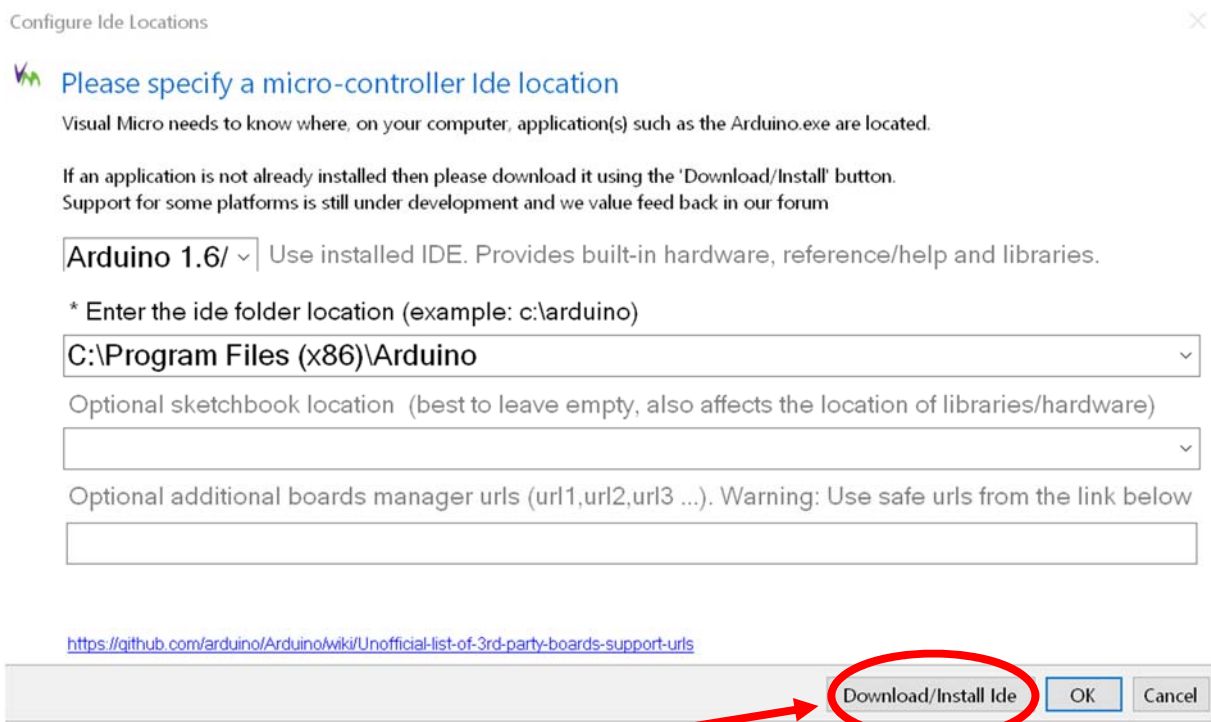


„Ohne Code fortfahren“

Nun erscheint in der Menüleiste von Visual Studio die Auswahl „vMicro“. Dies ist die Erweiterung für die Arduino-Entwicklung. Es fehlt jedoch hier noch die Funktionsschnittstelle zur ARDUINO – Hardware. Diese erhalten wir durch Aufruf der Installation:



Hier klicken:



„Download/Install IDE“

Die nun folgende ARDUINO-Oberfläche kann sich je nach Kreativität der Arduino-Leute auch mal ändern. Scrollen in der Seite, bis soetas ähnliches erscheint:

Klick „Download/Install Ide“

Die darauffolgenden ARDUINO-Seiten schauen je nach Versionsentwicklung ein wenig anders aus. Hier die Screenshots der gerade aktuellen Version:



Download the Arduino IDE



Windows Installer u. den Installer starten (Defaults übernehmen)



„Just Download“

Dies installiert auch die erforderlichen USB-Treiber

Die Original Arduino-Umgebung wird installiert:

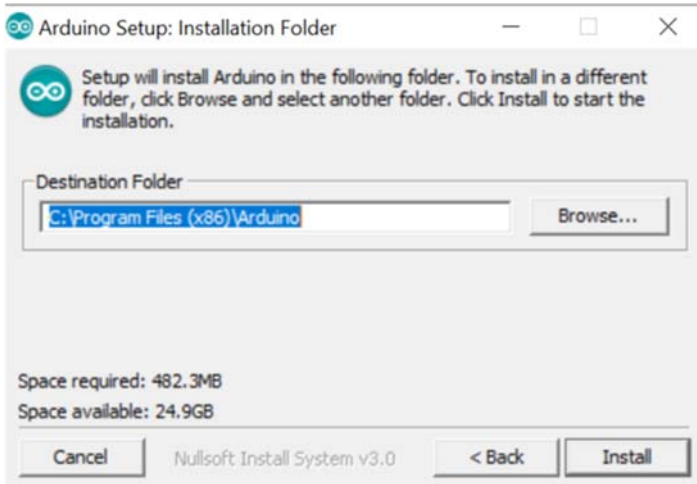
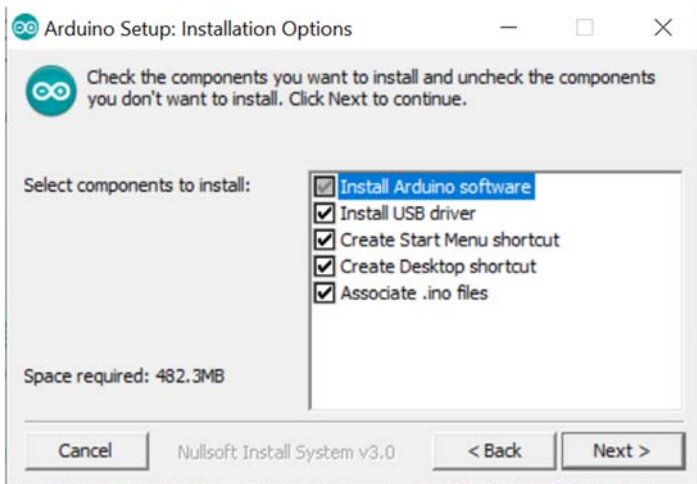
(öffnet arduino.cc im Web-Browser und bietet die aktuelle Version zum Download an. Wer bezahlen will klickt auf den Button ... wer nicht, klickt daneben „Just Download“)

Oder je nach Geschmack:

Zip-File entpacken, oder self Installer starten, je nach dem.

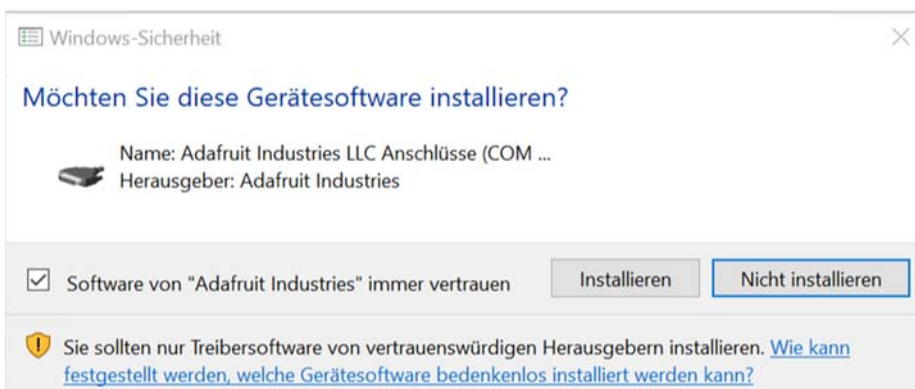
Self installer „Ausführen“

exe-Starten, EULA bestätigen, die vorgeschlagenen Komponenten übernehmen:



Hier nochmals das zuvor angelegte Installationsverzeichnis überprüfen. Dies muss mit der Angabe s.o. zwingend übereinstimmen. Wenn nicht dann entsprechend hier korrigieren („Browse“)

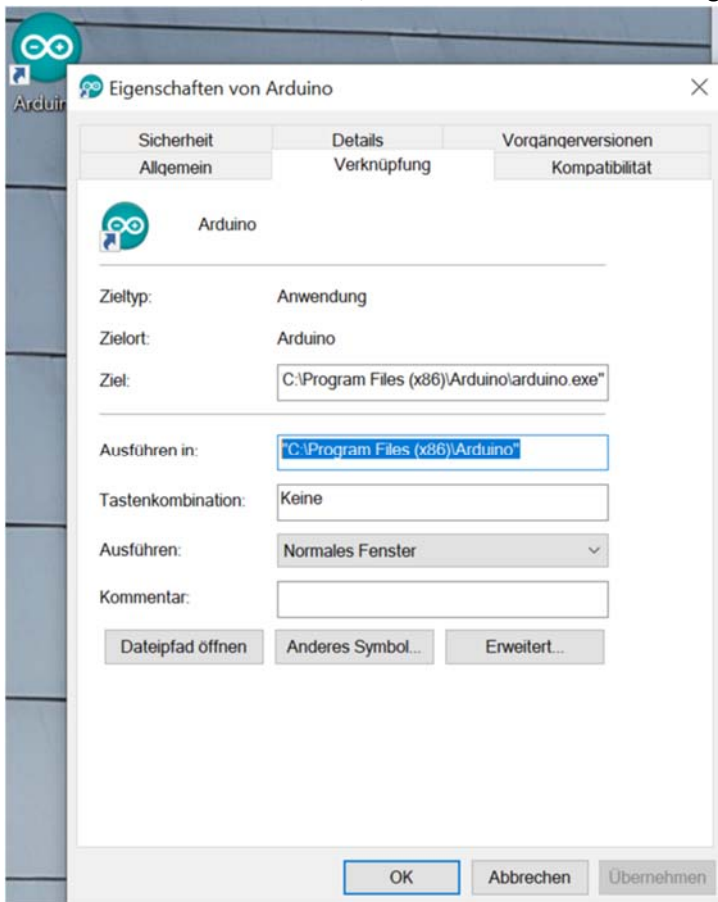
Das vorgeschlagene Verzeichnis übernehmen, dies wird in der Visual Micro Installation (s.o.) angegeben. Die vorgeschlagene Gerätesoftware am besten installieren.



Und evtl. weitere USB-Treiber bestätigen, weil diese später benötigt werden für die Datenverbindung
Web.Browser schließen. Die erscheinende Webseite kann man sich evtl. als Fav. Sichern
Visual Studio nun schließen.

Wenn man die ARDUINO DIE unbedingt benutzen möchte, was eigentlich nicht nötig ist, denn wir haben ja nun im Visual Studio eine leistungsfähige Entwicklungsumgebung, dann sollte der Vollständigkeit halber das Eigenschaftsfenster kontrolliert werden, um keine unangenehmen Überraschungen zu erleben.

Auf dem Desktop erscheint der Arduino Programm-Button:
Die Eigenschaften anzeigen lassen und im Fenster „Verknüpfungen“ den Eintrag „Ausführen in“ überprüfen.
Wenn das Feld leer sein sollte, dann bitte hier das zuvor angelegte Verzeichnis eintragen



„Ausführen in“ markieren, kopieren, und hier einsetzen. (ohne Hochkomma)

#####

ENDE Teil 1

#####