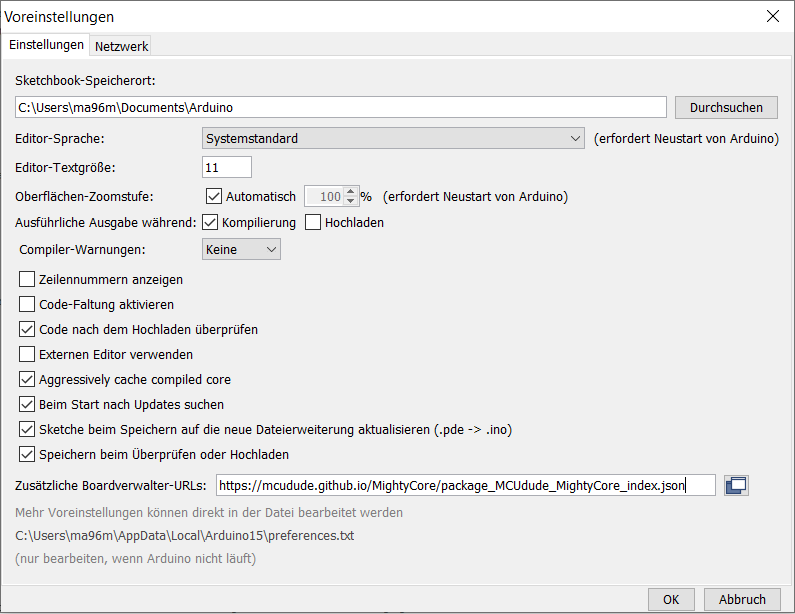
ATMega1284p mit Arduino programmieren

Da der ATMega1284p nicht auf einem Arduino-Board erhältlich ist, lässt er sich mit der Simulationssoftware Proteus nur in C oder Assembler programmieren. Für den realen Baustein gibt es die Möglichkeit, den sogenannten [Mighty Core](https://github.com/MCUdude/MightyCore) herunterzuladen, welcher dann eine Programmierung von diversen PDIP Varianten der ATMega Reihe in Arduino ermöglicht. Ein kleiner Nachteil ist, dass man das Programm in der Arduino IDE schreiben und das .hex, bzw. .elf File in Proteus verlinken muss, um den µC zu simulieren.

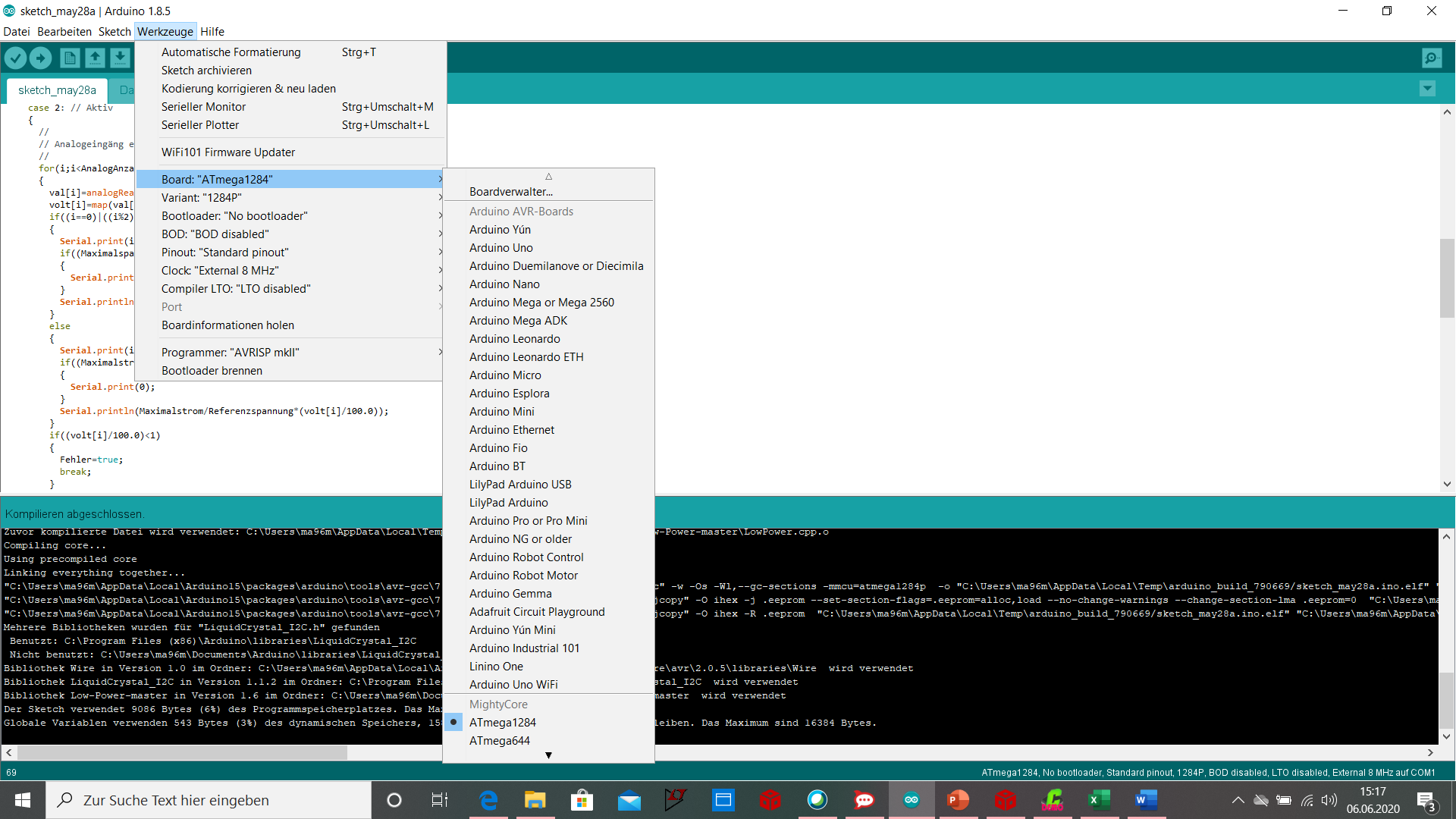
Einbinden des Mighty Cores

Unter Datei -> Voreinstellungen muss das Git Projekt als zusätzlicher Boardverwalter verlinkt und ein Haken bei Kompilierung gesetzt werden:

<https://mcudude.github.io/MightyCore/package_MCUdude_MightyCore_index.json>



Im nächsten Schritt, muss unter Werkzeuge -> Board der entsprechende Baustein unter Mighty Core ausgewählt und die entsprechenden Parameter, wie Oszillator, Bootloader, etc. ausgewählt werden.

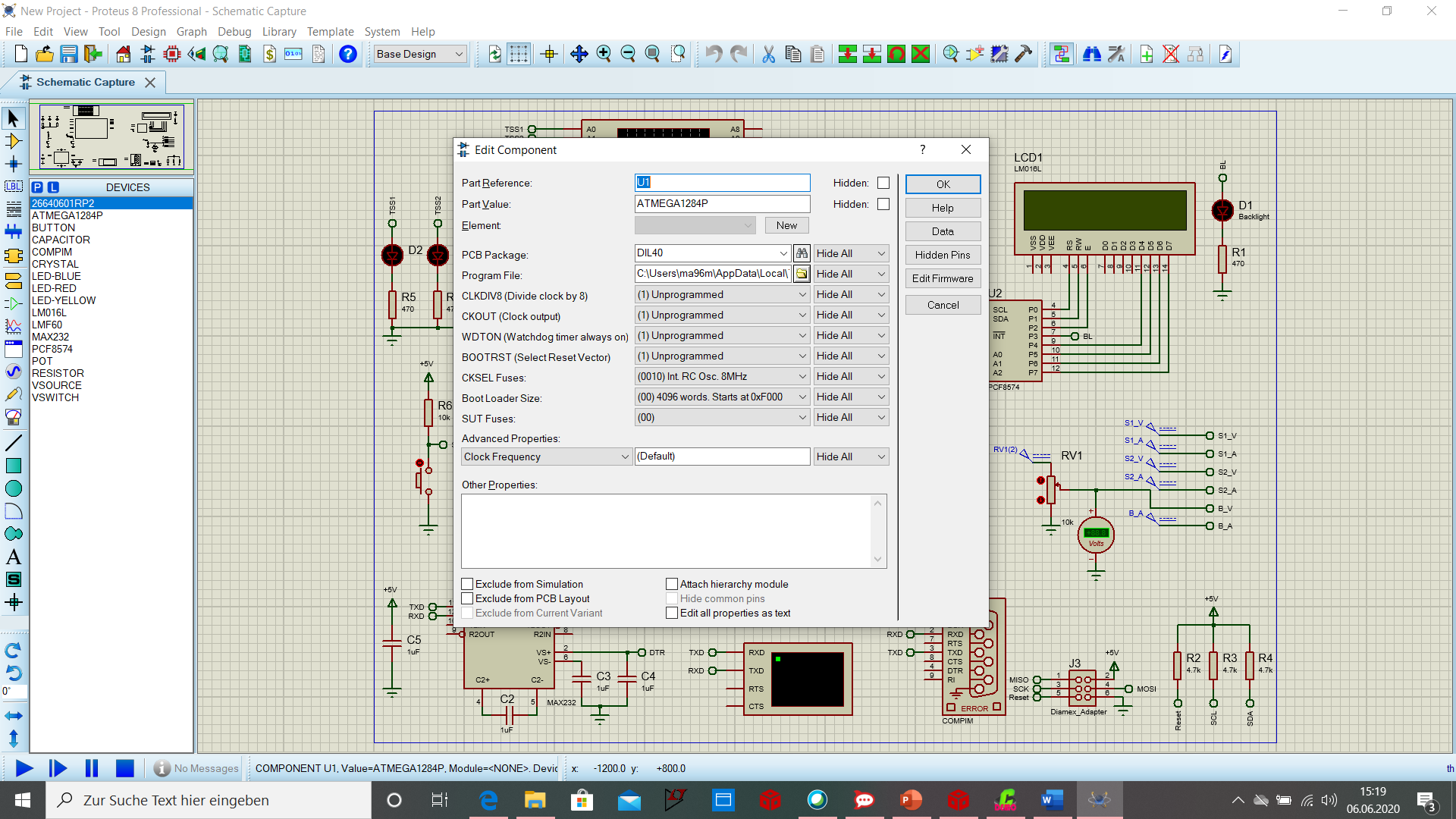


Der Baustein kann nun wie jeder andere Arduino programmiert werden. Ist das Programm fertig, wird mit Sketch -> Überprüfen/Kompilieren (oder dem Haken oben links im Schnellzugriff) kompiliert und die .hex, bzw. .elf Datei in den µC auf Proteus verlinkt. Der Pfad zu diesen Dateien ist im Output-Fenster nach erfolgreicher Kompilierung zu finden.





In Proteus wird nun der gewünschte µC in die Simulation eingefügt und die Einstellungen des Bausteins geöffnet. Und das .hex, bzw. .elf File als Program-File eingefügt.



Nun kann die Simulation ganz normal gestartet werden.