

VEX EDR – Einstieg leichtgemacht

Für einen schnellen Einstieg in die VEX EDR Welt sind im Folgenden verschiedene im Internet verfügbare Inhalte aufgeführt.

Für die Inhalte der Websites und der Downloads sind die Betreiber der jeweiligen Website verantwortlich.

Herstellerinformationen

- 🌐 Vielfältige Informationen zu allen Themen direkt vom Hersteller:
<https://www.vexrobotics.com/vexedr/products>

Aufbaubeispiele



- 🌐 Mit den VEX EDR Bausätzen können diverse Roboter aufgebaut werden. Einige **Beispiele** mit den dazugehörigen **Bauanleitungen** sind auf der Internetseite von VEX Robotics zu finden:
<https://www.vexrobotics.com/vexedr/support/robot-builds>

Online-Hilfe und Foren im Internet

- 🌐 VEX EDR Support – die Übersicht zu den Hilfethemen von VEX Robotics:
<https://www.vexrobotics.com/vexedr/support>
- 🌐 **VEX Robotics Knowledge Base** (englisch) – Tipps & Tricks zum VEX EDR direkt vom Hersteller:
<https://help.vex.com/>
- 🌐 **VEX EDR Forum** (englisch) – das offizielle Forum rund um den VEX EDR:
<https://www.vexforum.com/>

VEX EDR Videos

- 🌐 **VEX EDR V5 Produkt Video** (YouTube-Video):
<https://www.youtube.com/watch?v=HlJmc5k-2TU>
- 🌐 **VEX EDR V5 Controller System** – Einblick in den Lieferumfang des neuen V5 mit detaillierten Erläuterungen (YouTube-Video):
<https://www.youtube.com/watch?v=M35FiiCzoZs>
- 🌐 **Programmieren mit dem VEX Coding Studio** (YouTube-Playlist mit 4 Videos):
<https://www.youtube.com/watch?v=9-enZx6ePII&list=PLvvcc7S26YEjxOD7eCBuy7kJcebvrF3B>
- 🌐 Eindrücke von der **VEX Robotics Competition „Turning Point“** (YouTube-Video):
<https://www.youtube.com/watch?v=CDDGBcs0TFM>

Anleitungen und Unterrichtsmaterialien

- 🌐 Die **VEX EDR STEM Labs** sind eine Reihe von Unterrichtsprojekten sortiert nach Niveau und fachlicher Ausrichtung. STEM (science, technology, engineering, math) ist das englische Pendant zu MINT und fokussiert exakt auf dieselben fächerübergreifenden Möglichkeiten, die Robotik-Sets zu einem umfassenden didaktischen Baukasten machen.

Die Unterrichtsprojekte sind als HTML-, PDF- und PowerPoint-Versionen in englischer Sprache verfügbar:

<https://education.vex.com/eduvex/edr/stem-labs/>

Firmware-Update

Zur einwandfreien Funktion des VEX EDR V5 Robotik-Bausatzes sollte die Firmware VEXos auf dem aktuellen Stand sein.

Das Firmware-Update erfolgt nach automatischer Prüfung bei Verwendung des Programmiertools VEX Coding Studio (siehe unten) oder durch Verwendung der separaten VEXos Utility Software.

- 🌐 Download und Beschreibung der **VEXos Utility Software** auf der VEX Robotics Website:
<https://www.vexrobotics.com/vexedr/products/firmware>

Programmieroberflächen

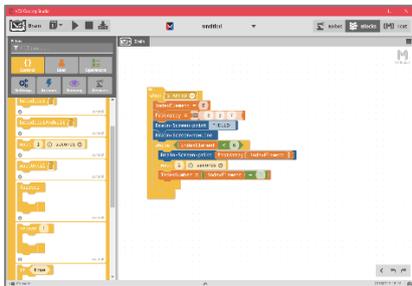


VEX Coding Studio – VCS

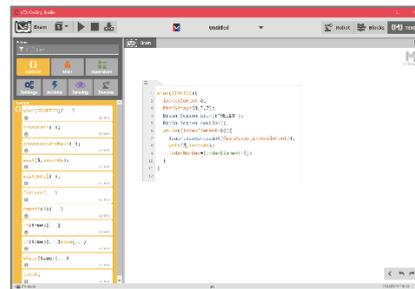
VCS ist eine kostenfreie Programmierumgebung für VEX IQ und VEX EDR V5 auf Desktop-Computern und Tablets. Die grafische Oberfläche Modkit Blocks ermöglicht einen intuitiven Einstieg in die Programmierung und kann in Modkit Text lückenlos textuell weitergeführt werden. VEX C++ und C++ Pro erweitern schrittweise die Möglichkeiten und die Komplexität. In Vorbereitung sind Oberflächen für Python und JavaScript.



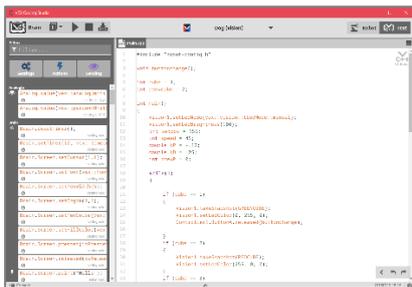
Modkit Blocks ist intuitiv zu bedienen und daher für Anfänger bestens geeignet. Diese Oberfläche basiert auf der bekannten grafischen Scratch Programmiersprache. VCS enthält diverse Beispielprogramme, die den Einstieg in die Programmierung noch einfacher machen.



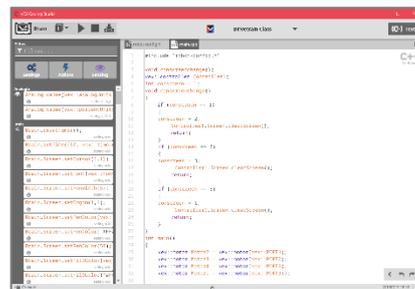
Modkit Text ermöglicht den einfachen Einstieg in die textuelle Programmierung. Erstellter Code kann beliebig zwischen Modkit Text und Blocks gewandelt werden.



VEX C++ ist eine rein text-basierte Oberfläche mit vielen hilfreichen Funktionen, wie Syntax-Kontrolle, Auto-Vervollständigung und farbigem Hervorheben.



C++ Pro bietet alle Möglichkeiten einer professionellen C++ Programmierumgebung. Alle Funktionen von VEX C++ sind integriert und sämtliche Beschränkungen aufgehoben.

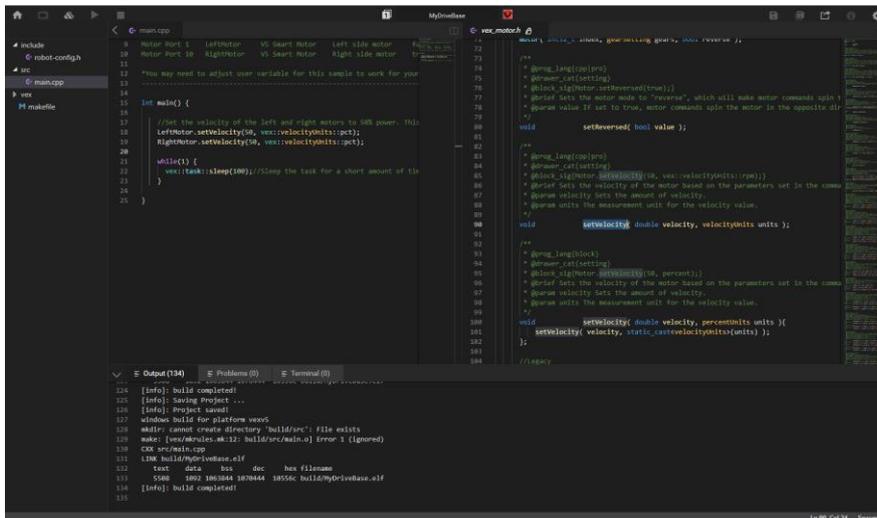


- 🌐 Infos zum **VEX Coding Studio** von VEX Robotics:
<https://www.vexrobotics.com/vexedr/products/programming>
- 🌐 Direkter Download-Link für **Windows-PCs**:
<https://link.vex.com/downloads/vcs-pc>
- 🌐 Direkter Download-Link für **macOS**:
<https://link.vex.com/downloads/vcs-mac>
- 🌐 Übersicht und Beschreibung der VCS Befehle in der **API Command Reference**:
<https://help.vexcodingstudio.com/>
- 🌐 Video-Anleitung zum **Programmieren mit dem VEX Coding Studio** (YouTube-Playlist mit 4 Videos):
<https://www.youtube.com/watch?v=9-enZx6ePII&list=PLVvcc7S26YEjxOD7eCBuy7kJcebvrrF3B>



VEXcode

VEXcode ist eine kostenfreie C/C++ Programmierumgebung ohne grafische Aufbereitung. Ideal für erfahrene Programmierer, die alle Funktionen ausnutzen wollen. Verfügbar für Windows und macOS, eine Chromebook Version ist genauso in Vorbereitung wie die Implementierung von Python und JavaScript.

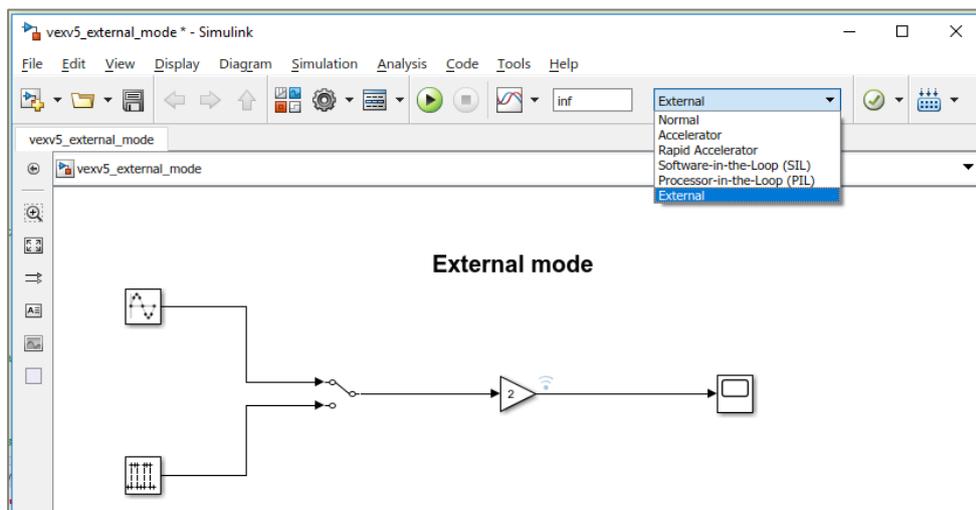


- 🌐 Beschreibung und Hilfeseiten zu **VEXcode**:
<https://www.vexforum.com/t/vexcode-preview/58625>
- 🌐 Direkter Download-Link für **Windows-PCs**:
<https://link.vex.com/vexcode-preview-pc>
- 🌐 Direkter Download-Link für **masOS**:
<https://link.vex.com/vexcode-preview-mac>



MATLAB & Simulink

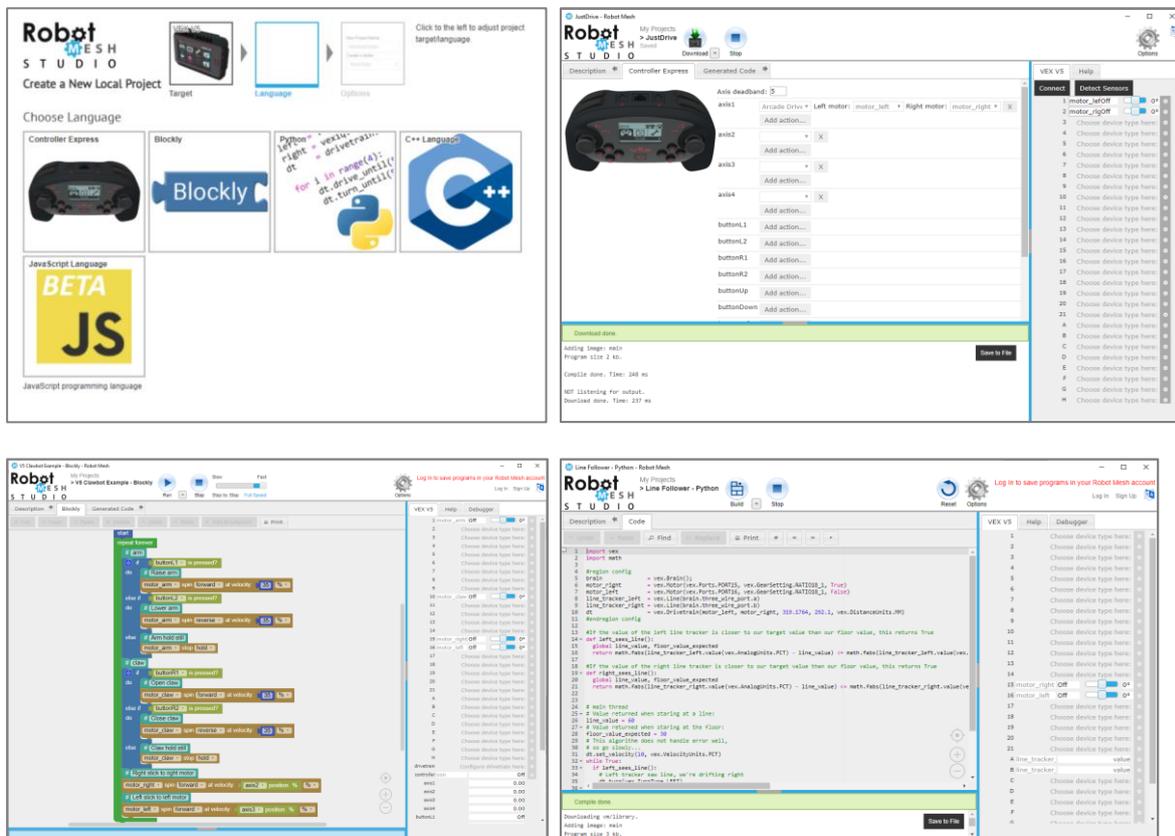
MathWorks® stellt für Projekte mit dem VEX EDR V5 die Software MATLAB® und Simulink® zusammen mit einer Anleitung, Kursmaterial und Programmierbeispielen für Bildungseinrichtungen kostenlos zur Verfügung.



- 🌐 Alle Infos dazu sind auf der **MathWorks®** Website zu finden:
https://www.mathworks.com/academia/student-competitions/vex-robotics.html?s_tid=srchtitle

Robot Mesh Studio

Robot Mesh Studio ist eine kostenfreie Programmierumgebung für VEX IQ und VEX EDR. Für den VEX EDR V5 werden 5 verschiedene Sprachen zur Programmierung angeboten: „Controller Express“, „Blockly“, Python, C++ und JavaScript.



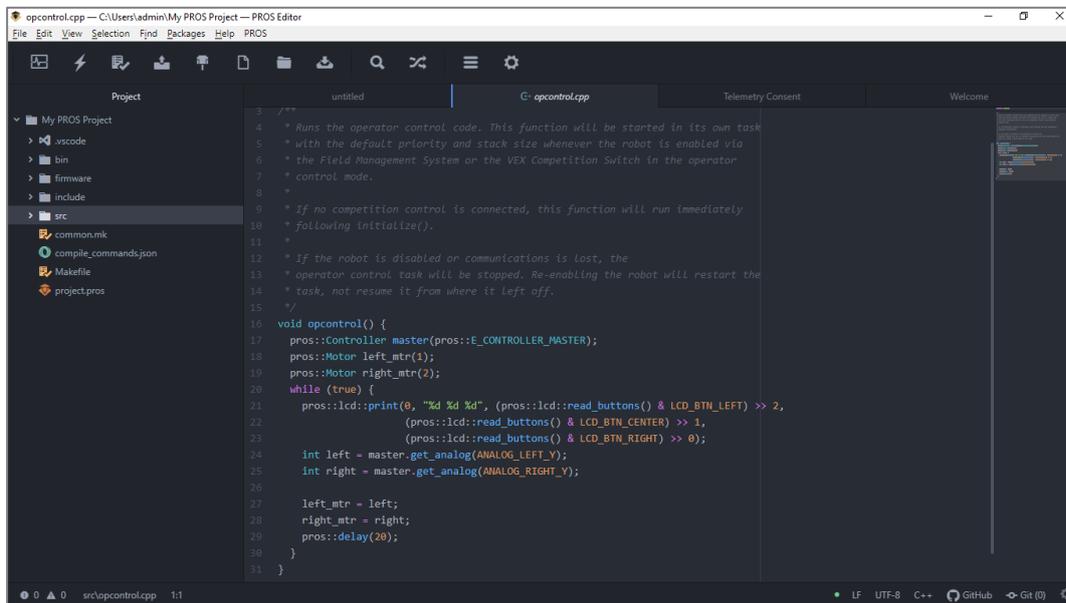
Für Windows-PCs ist eine Offline-Version mit msi-Installer verfügbar. Die Online-Version erlaubt das Programmieren über ein Browser-Plug-In auf **Windows**, **macOS X**, **Ubuntu Linux** und **Google Chromebook** und bietet darüber hinaus noch die Möglichkeit, virtuelle Roboter zu programmieren. Damit kann jeder Schüler selbstständig arbeiten und bekommt eine virtuelle Rückmeldung zur Qualität des eigenen Codes.

Die kommerzielle Education Edition enthält eine Benutzerverwaltung, Lizenzen für Flowol 4 und aufbereitete Lehrmaterialien. Erhältlich sind Jahres- und Dauerlizenzen für unterschiedliche Gruppengrößen.

- 🌐 **Robot Mesh Homepage:**
www.robotmesh.com
- 🌐 **Robot Mesh Studio** zur Online-Programmierung im Browser (Plug-In installieren!):
<http://rm.studio>
- 🌐 **Robot Mesh Studio Destop** als msi-Installer für Windows:
<https://store.robotmesh.com/desktop>

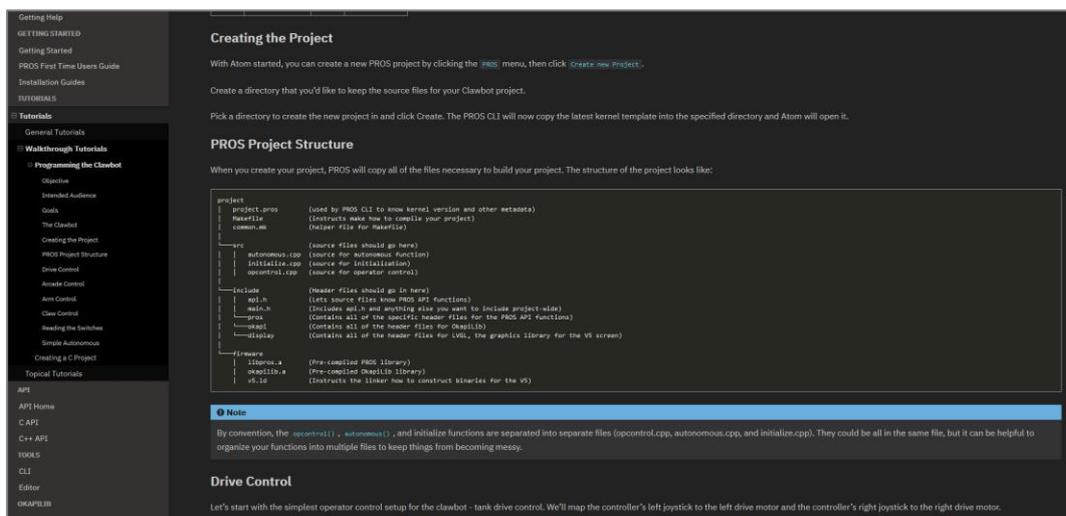
Pros

Pros nutzt den Open-Source-Texteditor Atom als flexible C/C++ Entwicklungsumgebung und stellt dafür ein Plug-In zur Verfügung. Pros kann somit auf Windows, macOS und Debian Linux Systemen verwendet werden. Übersichtlich erstellte Tutorials und eine ausführliche Beschreibung der Funktionen helfen bei den ersten Schritten.



```

3 //
4 * Runs the operator control code. This function will be started in its own task
5 * with the default priority and stack size whenever the robot is enabled via
6 * the Field Management System or the VEX Competition Switch in the operator
7 * control mode.
8 *
9 * If no competition control is connected, this function will run immediately
10 * following initialize().
11 *
12 * If the robot is disabled or communications is lost, the
13 * operator control task will be stopped. Re-enabling the robot will restart the
14 * task, not resume it from where it left off.
15 */
16 void opcontrol() {
17     pros::Controller master(pros::E_CONTROLLER_MASTER);
18     pros::Motor left_mtr(1);
19     pros::Motor right_mtr(2);
20     while (true) {
21         pros::lcd::print(0, "%d %d %d", (pros::lcd::read_buttons() & LCD_BTN_LEFT) >> 2,
22             (pros::lcd::read_buttons() & LCD_BTN_CENTER) >> 1,
23             (pros::lcd::read_buttons() & LCD_BTN_RIGHT) >> 0);
24         int left = master.get_analog(ANALOG_LEFT_Y);
25         int right = master.get_analog(ANALOG_RIGHT_Y);
26
27         left_mtr = left;
28         right_mtr = right;
29         pros::delay(20);
30     }
31 }
    
```



Creating the Project

With Atom started, you can create a new PROS project by clicking the **PROS** menu, then click: **Create new Project...**

Create a directory that you'd like to keep the source files for your Clawbot project.

Pick a directory to create the new project in and click **Create**. The PROS CLI will now copy the latest kernel template into the specified directory and Atom will open it.

PROS Project Structure

When you create your project, PROS will copy all of the files necessary to build your project. The structure of the project looks like:

```

project
├── project.pros      (used by PROS CLI to know kernel version and other metadata)
├── Makefile         (instructions on how to compile your project)
├── common.mk       (header files for Makefile)
├── src              (source files should go here)
│   ├── autonomous.cpp (source for Autonomous Function)
│   ├── initialize.cpp  (source for initialize() function)
│   └── opcontrol.cpp   (source for operator control)
├── include         (header files should go in here)
│   ├── api.h        (lets source files know PROS API functions)
│   ├── math.h       (includes math and anything else you want to include project-wide)
│   ├── pros         (contains all of the specific header files for the PROS API functions)
│   ├── pros_api     (contains all of the header files for pros_api)
│   └── delay         (contains all of the header files for delay)
├── firmware        (pre-compiled PROS library)
├── firmware.a      (pre-compiled PROS library)
└── v5.ld           (instructs the linker how to construct binaries for the V5)
    
```

Note
By convention, the `opcontrol()`, `autonomous()`, and `initialize` functions are separated into separate files (`opcontrol.cpp`, `autonomous.cpp`, and `initialize.cpp`). They could be all in the same file, but it can be helpful to organize your functions into multiple files to keep things from becoming messy.

Drive Control

Let's start with the simplest operator control setup for the clawbot - tank drive control. We'll map the controller's left joystick to the left drive motor and the controller's right joystick to the right drive motor.

Pros ist im Rahmen eines studentischen Projektes an der Purdue University entstanden und ist unter den Bedingungen der Mozilla Public License 2.0 veröffentlicht.

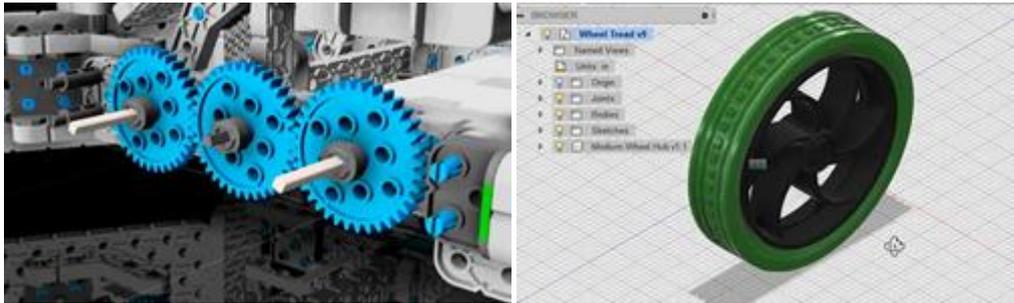
- 🌐 **Pros Homepage:**
<https://pros.cs.purdue.edu/>
- 🌐 **Download von GitHub:**
<https://github.com/purduesigbots/pros-cli/releases>

CAD-Software

Autodesk



Autodesk ermöglicht Bildungseinrichtungen den kostenfreien Zugang zu seinen professionellen Design-Programmen. Schüler, Studenten und Lehrer können verschiedene Autodesk Softwareprodukte nutzen und an Schulungen teilnehmen. Eine Bibliothek mit VEX Bauteilen wird zur Verfügung gestellt. Darüber hinaus gibt es umfassendes Schulungsmaterial.



- 🌐 **Kostenfreies Software-Angebot** für Bildungseinrichtungen von Autodesk:
<http://www.autodesk.de/education/>
- 🌐 **Autodesk Bibliotheken** für VEX IQ zum Download von der Autodesk website (für Inventor „Inventor_2013_VEX_KOP-Metric.zip“; für Fusion 360 „Fusion_360_VEX_IQ_KOP.zip“):
<http://www.autodesk.com/education/competitions-and-events/vex/recommended-software#Kit-of-parts>
- 🌐 **Projektbeispiel:**
<https://gallery.autodesk.com/projects/135845/case-pro-v5-fie---vex-robotics-and-recfoundation>

STEP-Dateien

Als standardisiertes Format zum Austausch von CAD-Daten können STEP-Dateien von nahezu allen CAD-Programmen importiert werden.

- 🌐 **STEP-Dateien** können von der jeweiligen Produktseite des Bauteils heruntergeladen werden, z.B.:
https://www.vexrobotics.com/vexedr/products/v5-products/276-4810.html#CAD_Files