

Table of contents

- Open Core Interface

Open Core Interface

Open Core Interface

With "Open Core Interface for drives", Bosch Rexroth provides interfaces which allow for an easy connection of the machine automation to the higher-level IT automation.

For different programming environments, compatible function libraries are provided in an SDK (Software Development Kit). In these libraries, functions for direct access to all data and functions of the drive are contained.

"Open Core Interface for drives" provides the following options:

- **Smart devices**

By using the smart devices and the installed apps, new operating concepts for systems can be developed.

The apps allow for access to all hardware functions of the smart devices such as the camera or the acceleration sensors. They provide the entire control comfort of smart devices for commissioning, operation and generation of diagnostic messages of ctrlX DRIVE, IndraDrive and HydraulicDrive devices.

Open Core Interface for drives currently supports the Android, iOS and Windows Phone operating systems.

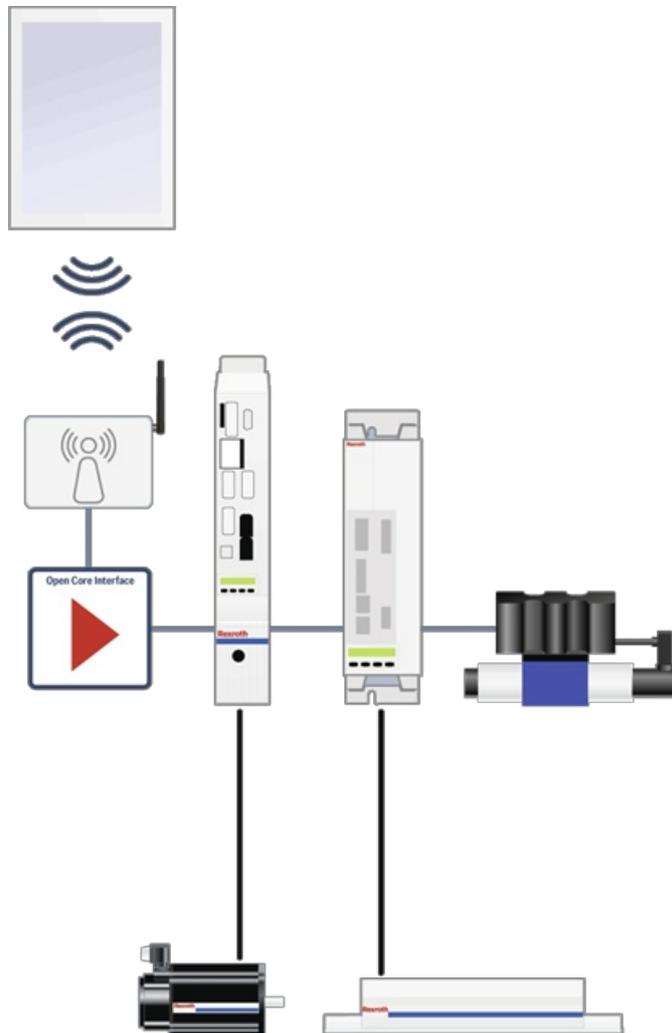


Fig. 142: Commissioning, operation and diagnostics of ctrlX DRIVE, IndraDrive and HydraulicDrives via Smart devices

■ **IT automation**

Open Core Interface in the IT automation domain refers to the use of PC-based solutions in the automation environment of a production machine.

For fast data exchange, connections to Windows and Linux-based development environments are provided in which high-level languages such as Java, C, C++ and C# are used for programming.

■ **Rapid Control Prototyping**

Open Core Interface provides all options for Rapid Control Prototyping, a design method for the closed-loop and open-loop control development. It is used for the early development of processes without a real machine and thus considerably contributes to cost-optimized and risk-minimized development. Rapid Control Prototyping is supported by LabVIEW, Simulink™, MATLAB®.



Information and support regarding "Open Core Interface" is available in the [Forum](#). Here, you can also ask members and the Bosch Rexroth experts.

Target platforms

With "eal_sdk" ("Easy Automation Library"- "Software Development Kit"), Bosch Rexroth provides programming interfaces for the following target platforms:

- PC with Windows/PC-based control with LabVIEW
- PC with Linux as operating system
- (Mac OS upon request)
- Smart device with Android, iOS or Windows phone as operating system
- Linux-based controller (e.g., BeagleBone Black®, Raspberry pi®,...)

Supported devices

Using "eal_sdk", you can create applications for configuring and controlling Bosch Rexroth drive controllers and frequency converters. Devices supporting the "Sercos Internet Protocol" (S/IP) (devices with Multi-Ethernet interface) are supported:

- Electric drive controllers of the ctrlX DRIVE, IndraDrive Cs, IndraDrive C, IndraDrive M, IndraDrive Mi, IndraDrive ML ranges [with all control sections available in the product range (ECONOMY, BASIC and ADVANCED, single- and double-axis devices)]
- Hydraulic drive controllers "Integrated Axis Controller (IAC)", "Hydraulic Motion Control (HMC)"
- Frequency converters of EFC x610 type

Application options

One or several drives can be configured and controlled in star or line topology using the master communication interface.

Using "Open Core Interface for drives", the following applications can be used:

- "Open Core Interface for drives" can replace a conventional control unit. Both the master communication and the commanding of the drive controller are performed by "Open Core Interface for drives". The master communication has to be disabled for this use case.
- "Open Core Interface for drives" can be used in parallel with an active master communication (Sercos, PROFINET, EtherCAT with EoE, EtherNet/IP, Ethernet POWERLINK). In this case, the following applications may be implemented:
 - As HMI interface; commanding is realized via an external PLC via the master communication protocol.
 - As commissioning tool for loading a parameter file, replacing the drive firmware,...
 - As diagnostics and debugging interface by using the firmware oscilloscope.

If the master communication is active, it is not possible to switch the drive in control (AF) via the OCI interface.

SDK: Requirements and installation

To use the "eal_sdk", ("Easy Automation Library" "Software Development Kit"), please perform the following steps:

1. Make sure to use one of the following systems:
 - Electric drive controller with IndraDrive firmware MPx-18 (or higher)
 - Electric drive controller with ctrlX DRIVE Runtime AXS-V-02RS (or higher)
 - Hydraulic drive controller with HydraulikDrive firmware HDx-20 (or higher)
 - Frequency converters of EFC x610 type
 - IndraMotion MLD control system in version 13VRS or higher
2. For commissioning and engineering of the drives, install ctrlX DRIVE Engineering, IndraWorks DS/IndraWorks MLD in version 13VRS or higher. Alternatively, you can also use the "Drive tool" or "Drive tool EFC" application contained in the SDK.
3. **Registering for the Engineering Network**

To download the "eal_sdk" SDK from the Bosch Rexroth Internet page, you have to register for the **Engineering Network**:

 - Go to the internet page of **Bosch Rexroth**.
 - Register on "myRexroth." (For a description of the registration process, please refer to the **Engineering Network forum**.)
 - Open the webpage for "**Open Core Engineering**".
 - After the registration is completed, the SDK "eal_sdk" is available in the **Download area** of **Engineering Network**.
4. Depending on the application, the device platform and operating system and the target device, integrate the "Easy Automation Library" contained in the SDK in your development environment.

SDK: Contents

The SDK "eal_sdk" is divided into different "Tool boxes". In the SDK, each tool box has its own subdirectory. The functions supported by the "Tool boxes" are basically identical. They differ with regard to the development environment and the operating system in which the development environment is used.



Each to

olbox contains not only libraries, but also documentation and ↘ "Examples of application".

Supported functions of the tool boxes:

- **System**
 - Establish connection to the drive controller or the frequency converter
 - System information such as reading diagnostic data and firmware version
 - Firmware upload and download
- **Parameters**
 - Read and write parameters

- Read parameter name, attributes, unit and status
- Stopping and executing commands
- **Motion**
 - General functions (activating drive enable, bringing the drive to the STOP state, drive-controlled homing)
 - Axis motions (preset velocity, moving to absolute position, moving by a distance in addition to the target position, moving by a distance from the current actual position)
- **Oscilloscope**
 - Configuring the oscilloscope channels
 - Configuring the oscilloscope trigger
 - Reading the oscilloscope data

The SDK is continuously developed; the following chapters provide an overview of the "eal_sdk" SDK version 2.0



Since the development environments and operating systems may change, Bosch Rexroth does not assume any warranty for the proper functioning of the SDK.

EAL4Android

"EAL4Android" comprises an Android library and a "Mono for Android" app for configuring and controlling Bosch Rexroth drive controllers and frequency converters that support the "Sercos Internet Protocol" (S/IP).

System requirements: Java™ 32/64-bit, Android Studio

EAL4DotNet

The "EAL4DotNet" tool box allows functions of Bosch Rexroth drive controllers and frequency converters to be accessed by .NET libraries.

Supported operating systems of the target device: Windows 10

Supported development environment: Microsoft Visual Studio 2005 or newer, Visual Studio with Xamarin extension, Xamarin Studio, RAD studio®

Supported programming languages: .NET (C#, VB, F#), VBA (for MS Excel, MS Access, MS PowerPoint, MS Word), Delphi

EAL4C

Using the "EAL4C" tool box, the functions of "EAL4DotNet" are converted into structural functions with Mono.

Supported operating systems of the target device: Windows 10

Supported development environment: Eclipse, Microsoft Visual Studio, Qt, etc.

Supported programming languages: C/C++

EAL4Java

The "EAL4Java" tool box includes the "EAL4C" functionality as object-oriented Java classes in a package by using the native Java NDK library. EAL4Java allows EAL functions to be called from the Java runtime environment.



Applications using the "EAL4Java" tool box can only run on the Java runtime environment under Windows and Linux.

Supported operating systems of the target device: Windows 10

Supported development environment: Eclipse

Supported programming languages: Java

EAL4LabVIEW

The "EAL4LabVIEW" tool box allows functions of Bosch Rexroth drive controllers and frequency converters to be accessed under LabVIEW.

Supported operating systems of the target device: Windows 10

Supported development environment: National Instruments LabVIEW

Supported programming languages: G (LabVIEW VI language)

Examples of application

In the "Engineering Network" [[Download](#)→Apps (\ registration and login required)] and in the SDK, we provide different demo programs and apps.

For all demo programs and apps, the entire source code, the relevant project files, information about the use case, the requirements for execution and creation are provided.

In the following chapters, some examples are provided.

Windows application with Microsoft Visual Studio for controlling and monitoring "IndraDrive" and ctrlX DRIVE controllers

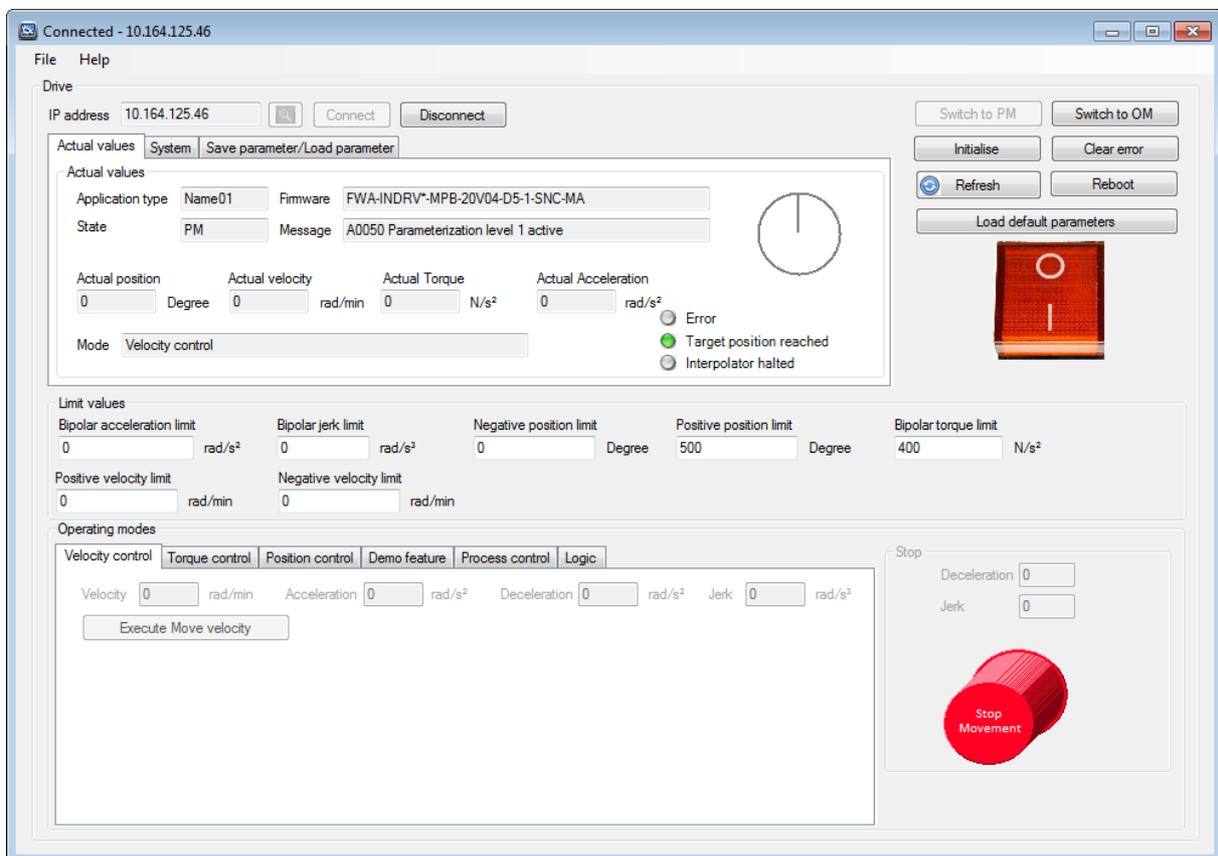
In the Microsoft Visual Studio development environment, the Windows application "DriveTool" is developed.



Under `..\eal4DotNET\Samples\Sample - DriveTool`, the "ea_sdk" SDK contains the "DriveTool" Windows application and also the source files.

"DriveTool" allows the following functions of "IndraDrive" range drive controllers to be accessed:

- Browsing available drives
- Establish connection to a drive via the Ethernet IP address
- Reading the power status and switching on power
- Restarting the drive
- Initializing the drive
- Reading the firmware type code
- Switching between operating mode and parameter mode
- Homing the drive
- Clearing drive errors
- Reading and selecting the operating mode
- Reading and writing the motor velocity, position and torque



DB001180v01.png

Fig. 143: "DriveTool"

Microsoft Excel file as user interface

The following example shows the integration of the "Open Core Interface for drives" in the Microsoft office Excel

application.

The Excel user interface allows the following functions to be accessed:

- Establish connection to a drive via the Ethernet IP address
- Switch on power
- Switching between operating mode and parameter mode
- Writing the motor velocity, acceleration, deceleration and jerk

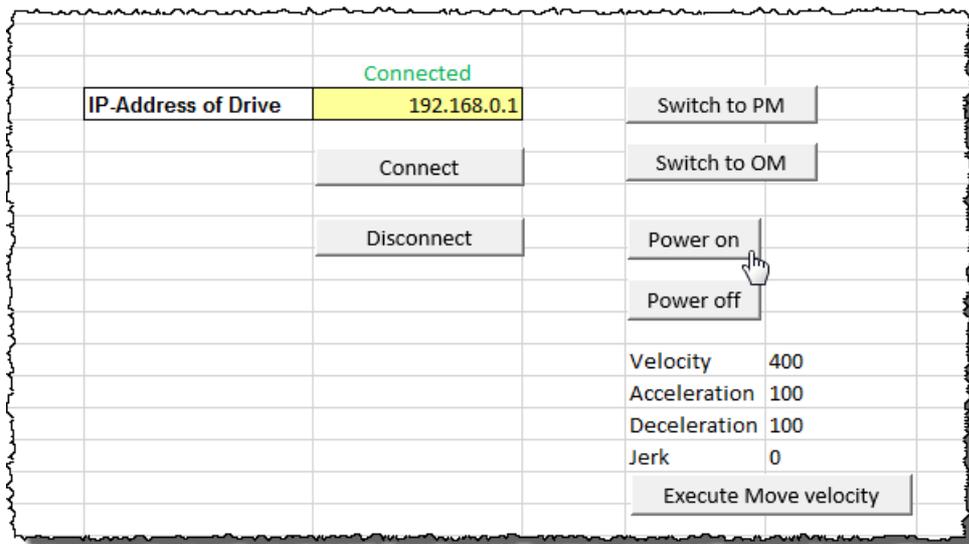


Fig. 144: Excel user interface

Windows application with Microsoft Visual Studio for recording device signals

In the Microsoft Visual Studio development environment, the Windows "Oscilloscope" application was developed.

In the application, you can configure, for example, the signal to be recorded, the sampling rate and the trigger to be used for commissioning, servicing and testing drive controllers.

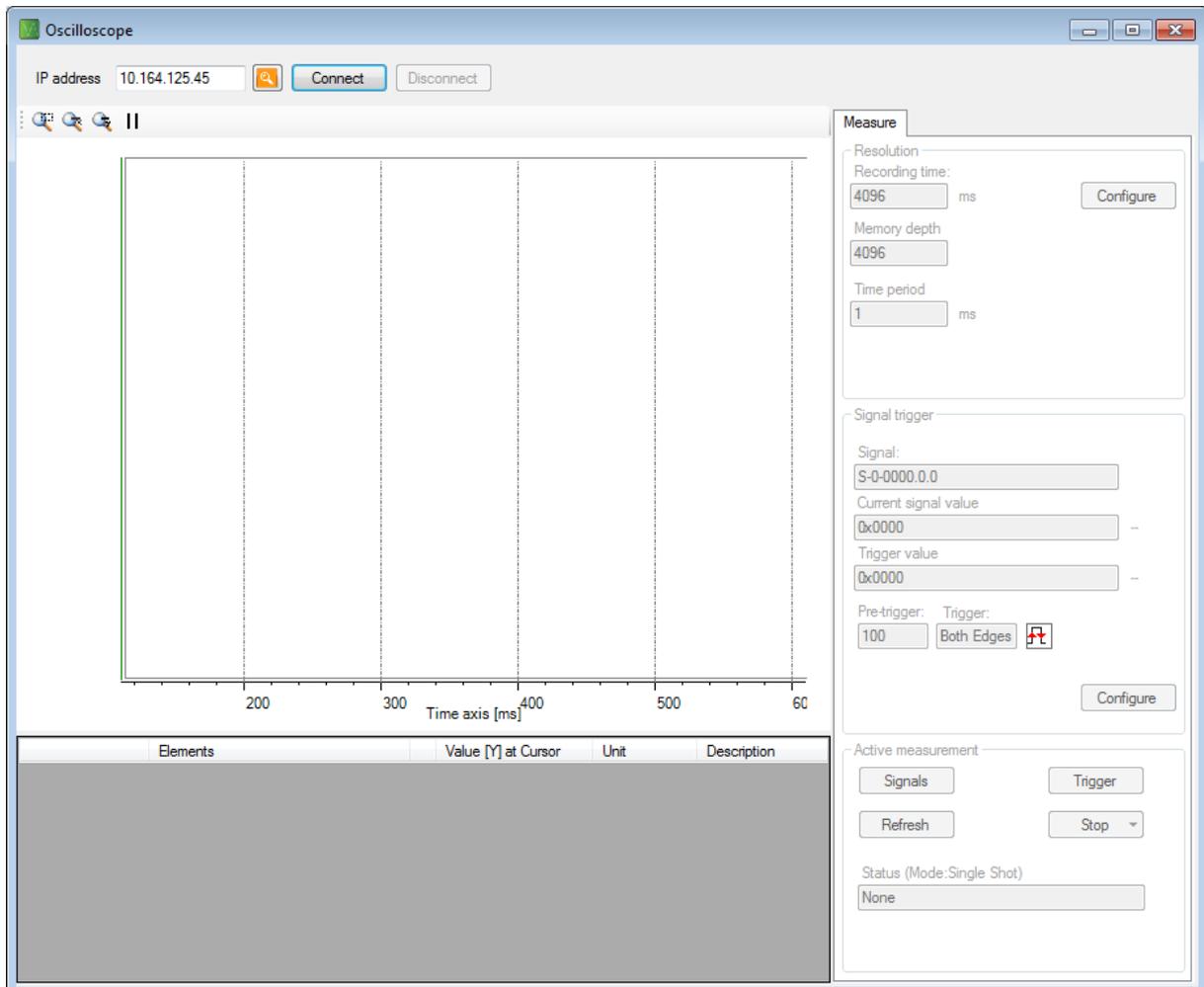


Fig. 145: Windows "Oscilloscope" application

Windows application with Microsoft Visual Studio for archiving and restoring firmware and parameter files

In the Microsoft Visual Studio development environment, the Windows "Download tool" application was developed.

Use the "Download tool" application to archive and restore firmware and parameter files of drive control devices of the "IndraDrive" product family.

The "Download tool" can be operated in console mode and in GUI mode with graphic user interface.

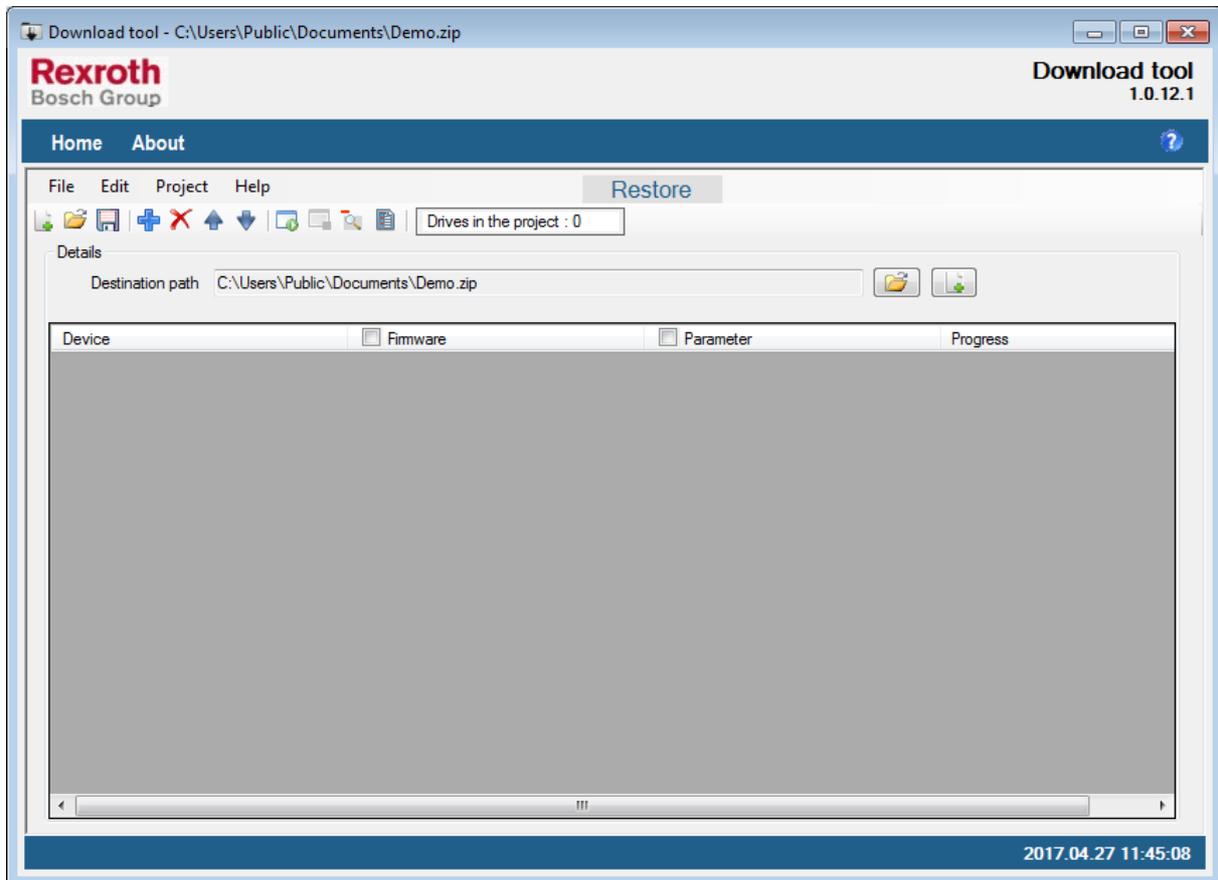


Fig. 146: Windows "Download tool" application in GUI mode with graphic user interface

Android app

For Android, different demo apps with different functionalities are provided in the SDK:

- Travel motion
- Oscilloscope function
- Parameter backup
- Firmware update
- ...

The demo apps are provided as APK (Android Package File).

To install the demo apps, you must permit the installation of apps not coming from the Google Play Store.

- Open the settings on your smart phone and set the check mark at *Applications → Unknown sources* (for some devices under *Safety → Unknown sources*).
CAUTION: The check mark is a safety risk since apps from unknown sources may contain viruses.
⇒ Only install apply of reliable sources.

⇒ By default, disable the check mark.

- The easiest option to transmit an APK from a computer to your smart phone is to use a USB cable.
- Transmission via Bluetooth is also possible. To do so, you have to rename the *.apk file extension before the transmission (e.g., *.txt).



Fig. 147: Transmitting SDK via Bluetooth

- After the transmission, the file has to be renamed *.apk on the smart phone.

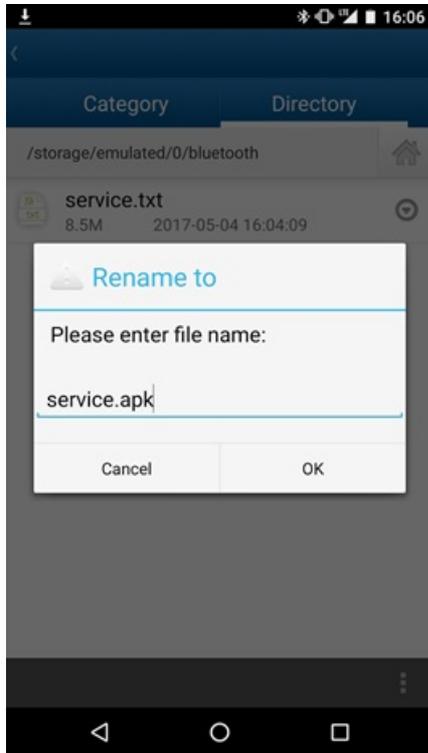


Fig. 148: Renaming SDK

- Install the APK on your smart phone.



To use "Open Core Interface for drives" on the smart phone, **you must in any case install the "service.apk" APK for each application APK.** "service.apk" is contained in the SDK in the "lib" subdirectory of the EAL4Android directory.

After the installation of "service.apk", you will find the app under [Settings → Applications](#) as "EAL Service".

Travel motion by entering command values

In the "Demo App" (demoapp.apk), the travel motions are performed by entering command values ("[MOVE VELOCITY](#)"/ "[MOVE FREQUENCY](#)"). Furthermore, limit values can be specified.

Here, the "Demo App" (demoapp.apk) with the steps "Establishing connection to the drive controller" and "Accessing the entry mask for velocity parameters":

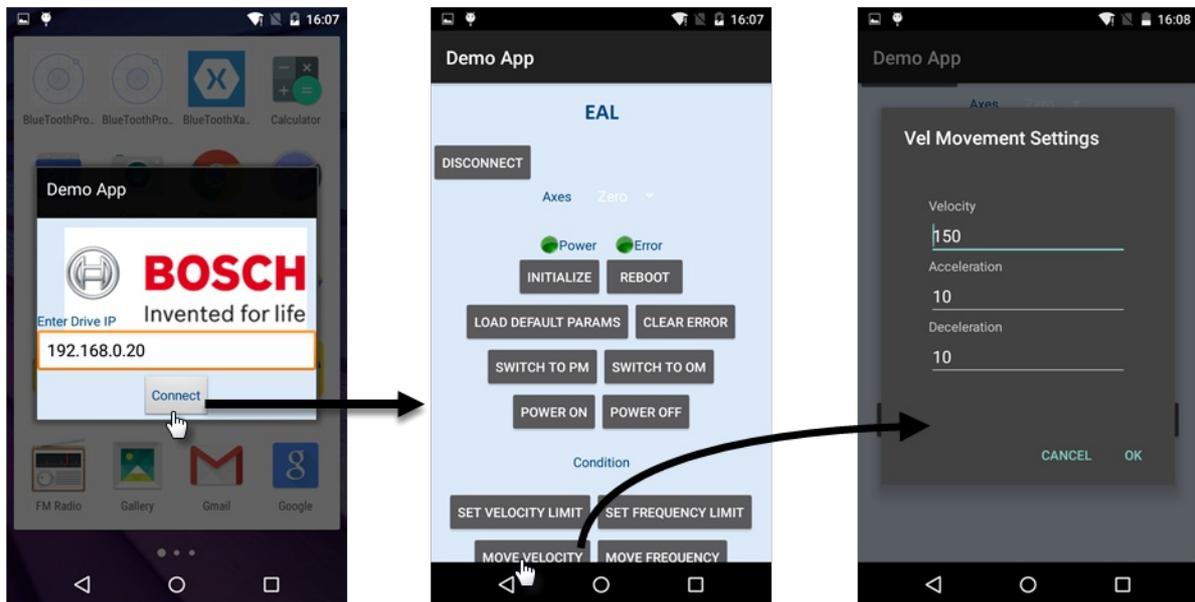


Fig. 149: "Establishing connection to the drive controller" and "Accessing the entry mask for velocity parameters" in the "Demo App"

Travel motion by rotary and tilting motion of the smart phone

To perform a travel motion of the drive, it is also possible to use the acceleration and position sensors of the smart phone. The figure below shows an app in which the drive is moved by rotary and tilting motions of the smart phone:

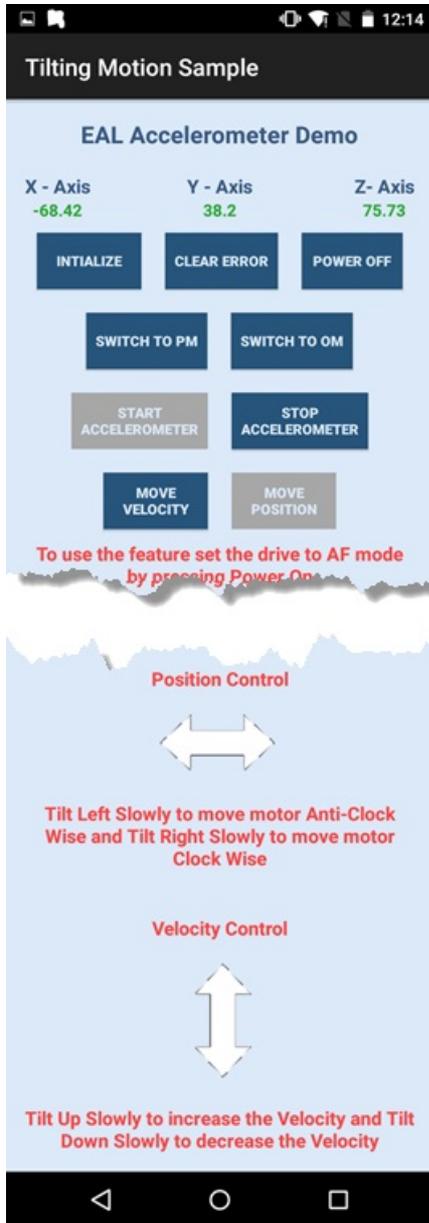


Fig. 150: Demo App: Travel motions by rotary and tilting motion of the smart phone