# Getting started with B-Box Micro

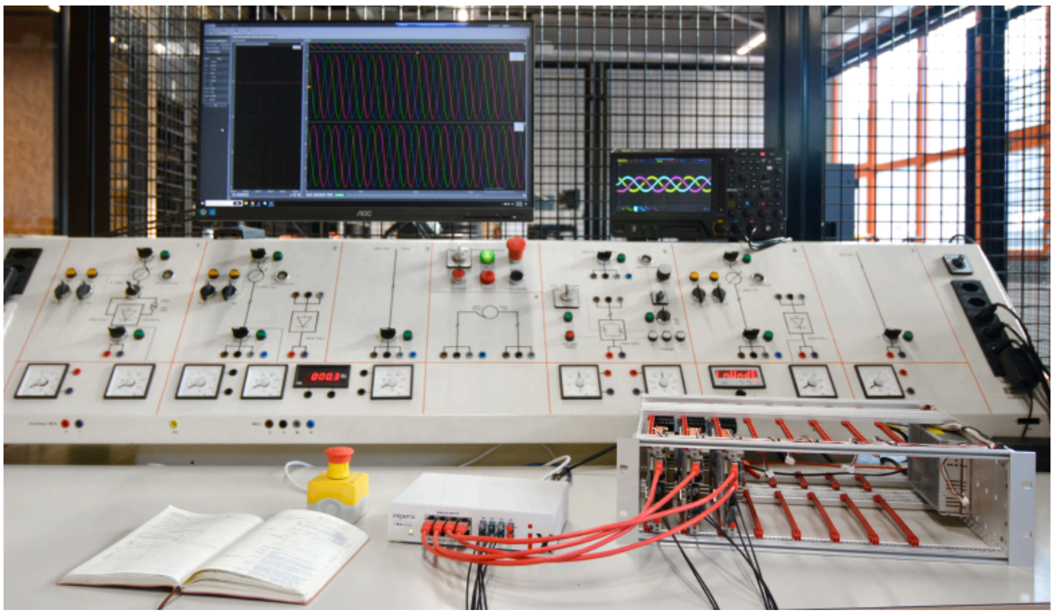
PN107 | Posted on February 15, 2024 | Updated on July 28, 2025

Daniel BLARDONE
Engineer
imperix • in

Table of Contents

This page delivers getting-started information for individuals new to operating the B-Box Micro. As a practical use case, a generic open-loop three-phase inverter application is implemented. Additionally, this page provides a comprehensive list of relevant examples that can be implemented with the B-Box Micro.

This article focuses on the seamless transition from simulation to lab testing with B-Box Micro rather than the theoretical aspects of the implemented three-phase inverter. The latter is addressed in this Three-phase inverter example.

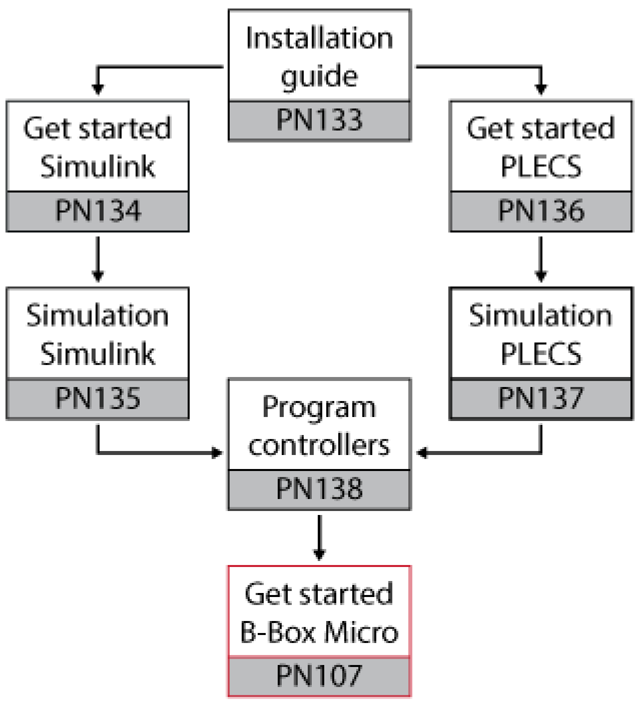


Experimental setup for an open-loop three-phase inverter

## Overview of useful resources

For a complete introduction to imperix software development kits, it is recommended to read the following articles beforehand:

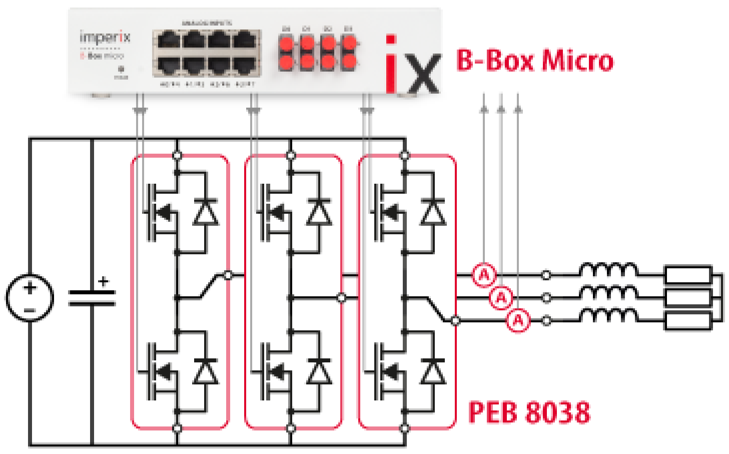
- **ACG SDK:**
  - Installation guide for the ACG SDK
  - **Simulink:**
    - Getting started with Simulink
    - Simulation essentials with Simulink
  - **PLECS:**
    - Getting started with PLECS
    - Simulation essentials with PLECS
- **CPP SDK:**
  - Installation and utilization of CPP SDK
- **PROGRAMMING IMPERIX CONTROLLERS:**
  - Programing and operating imperix controllers



## Implementing an experimental setup

The B-Box Micro is a tabletop controller specifically designed to facilitate the teaching of power electronics. The B-Box Micro can be programmed by writing C/C++ code (CPP SDK) or by drawing a block diagram in Simulink or PLECS (ACG SDK). The code is deployed onto the controller with the Cockpit software, using an Ethernet link. Imperix Cockpit also provides run-time interaction and monitoring from the host PC.

The illustration below shows the employed use case example, where the B-Box Micro controls a three-phase inverter.



Typical use case of the B-Box Micro
Open loop control of a three-phase inverter

This setup can be implemented with the imperix starter kit. Some additional material, listed below, is however required to test the system before operating the complete converter.

- Laboratory DC power supply
- 3x power resistors
- 3x 2.5mH inductor

During the presented experiments, the inverter is connected to a three-phase RL load, under the following conditions:

- DC-bus voltage: 100 V
- Control frequency: 20 kHz
- Sampling phase: 0.5 (middle of the switching period)
- Load resistance: 6 Ω
- Load inductance: 2.5 mH

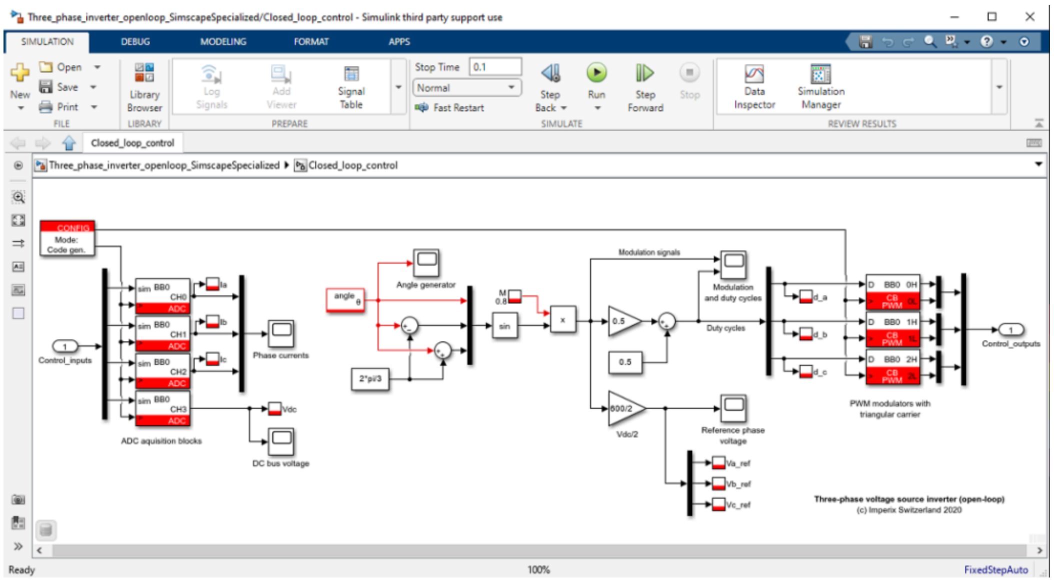
## How to program the B-Box Micro from Simulink

To develop and flash a run-time code using Matlab Simulink, the necessary software must be installed on a computer (the requirements and the procedure are similar using PLECS):

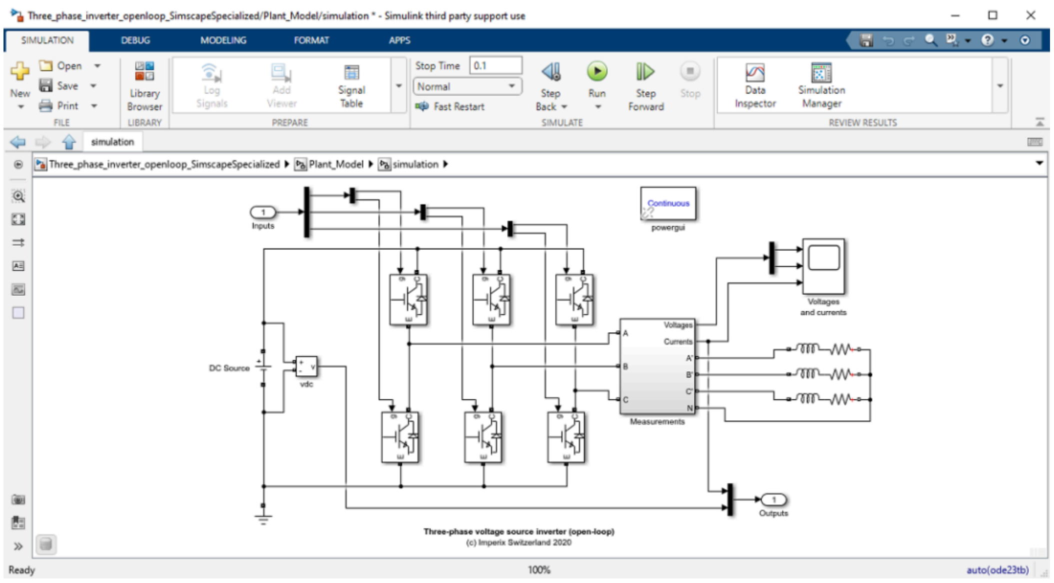
1. Main software: Matlab Simulink, with Matlab Coder, Embedded Coder, and Simulink Coder
2. Imperix ACG SDK
3. Optionally: a plant simulation software (e.g. Simscape Electrical or PLECS blockset for Simulink).

Support for the B-Box Micro has been added since version 2024.1 of the SDKs. When working with an older version of SDK, upgrading to 2024.1 is required. Downloads are available here.
**Licensing policy**
Imperix ACG SDK requires a paid license for loading the generated code on a programmable controller. The license gives unlimited access to the controllers. All software licenses are target-locked (i.e. usable on multiple computers) and lifetime (no renewal fees, free software updates).

Download **Three_phase_inverter_B-Box_Micro**


Three_phase_inverter_openloop_SimscapeSpecialized/Closed_loop_control - Simulink third party support use
CONFIG
Mode: Code gen.
Control_inputs
ADC aquisition blocks
Phase currents
DC bus voltage
Angle generator
Modulation signals
Modulation and duty cycles
Duty cycles
Reference phase voltage
PWM modulators with triangular carrier
Control_outputs
Three-phase voltage source inverter (open-loop)
(c) Imperix Switzerland 2020

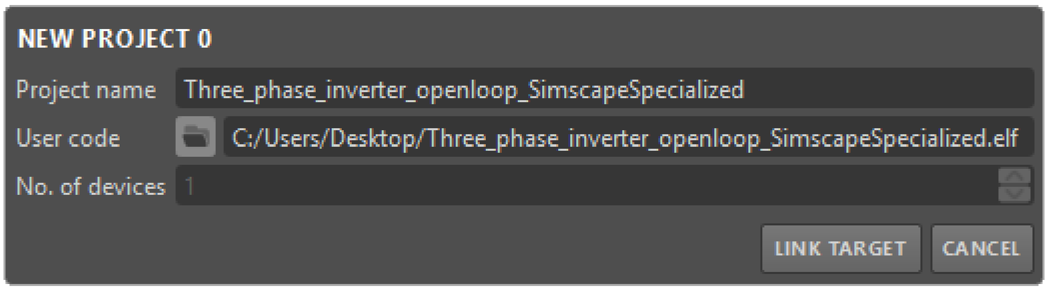
Control side of the VSI


Three_phase_inverter_openloop_SimscapeSpecialized/Plant_Model/simulation * - Simulink third party support use
Inputs
Continuous
powergui
DC Source
vdc
Voltages and currents
Measurements
Outputs
Three-phase voltage source inverter (open-loop)
(c) Imperix Switzerland 2020

Power side of the VSI

Once the model is opened, the code generation option must be selected in the "CONFIG" block. To generate code, click the "Build" button or use the shortcut (Ctrl+B in Simulink). Regardless of the workflow that is used, the build procedure is done such that everything takes place automatically. This comprises code generation, compilation, and upload of the user code onto the controller.

At the end of a successful build, Cockpit automatically launches and creates a new project with a pre-filled project name and executable file path. To load the user code (.elf file) to the target and start the code execution, click on "Link target" and choose your device using the Cockpit left bar. The user code is then automatically transferred to the B-Box.

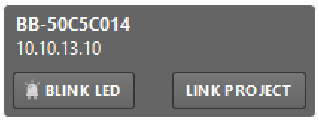

NEW PROJECT 0

Project name: Three_phase_inverter_openloop_SimscapeSpecialized

User code: C:/Users/Desktop/Three_phase_inverter_openloop_SimscapeSpecialized.elf

No. of devices: 1

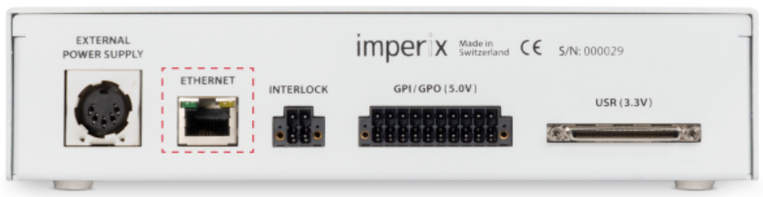
Cockpit creates a new project using the user code built from Simulink, PLECS, or imperix IDE.


BB-50C5C014
10.10.13.10

Upon clicking "Link target" Cockpit displays a list of potential targets to upload the user code to.

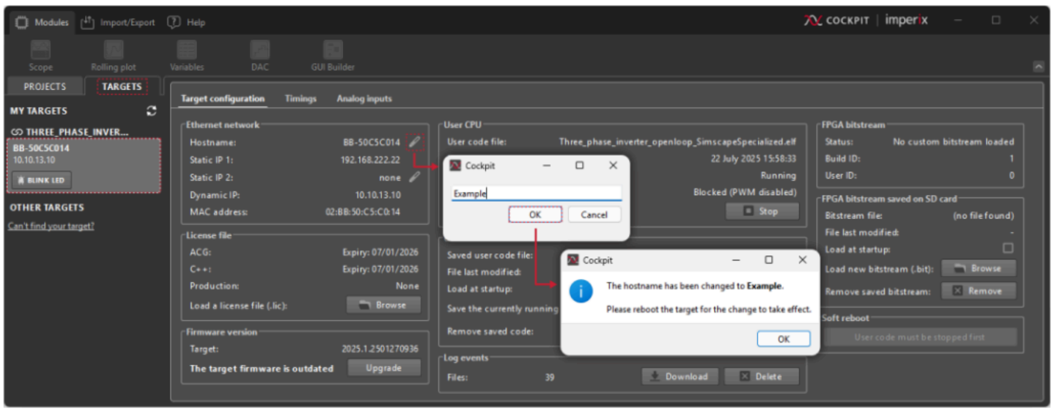
## How to connect to the B-Box Micro from the PC

The connection to the host computer is required to deploy the user code and to provide real-time access to the controller variables during run-time. The location of the ethernet port on the B-Box Micro is on the rear side, as highlighted in the following figure.



Location of the ethernet port on the rear side of the B-Box Micro

The B-Box Micro does not feature a display. In a regular classroom setup where different workstations are operated, the best way to address the lack of display is to rename the *hostnames* with the corresponding workstation numbers. By doing so, once the B-Box Micro is assigned to a specific workstation, there will be no need to verify the correct connection during future training sessions.



BB-50C5C014
10.10.13.10

Ethernet network
Hostname: BB-50C5C014
Static IP 1: 192.168.222.22
Static IP 2: none
Dynamic IP: 10.10.13.10
MAC address: 02:BB:50:C5:C0:14

License file
ACG: Expiry: 07/01/2026
C++: Expiry: 07/01/2026
Production: None
Load a license file (.lic):

Firmware version
Target: 2025.1.2501270936
The target firmware is outdated

User CPU
User code file: Three_phase_inverter_openloop_SimscapeSpecialized.elf
22 July 2025 15:58:33
Running
Blocked (PWM disabled)

Saved user code file:
File last modified:
Load at startup:
Save the currently running
Remove saved code:

Log events
Files: 39

FPGA bitstream
Status: No custom bitstream loaded
Build ID: 1
User ID: 0

FPGA bitstream saved on SD card
Bitstream file: (no file found)
File last modified: -
Load at startup: ☐
Load new bitstream (.bit):
Remove saved bitstream:

Soft reboot

Cockpit
Example

Cockpit
The hostname has been changed to **Example**.
Please reboot the target for the change to take effect.

How to change the hostname through Cockpit

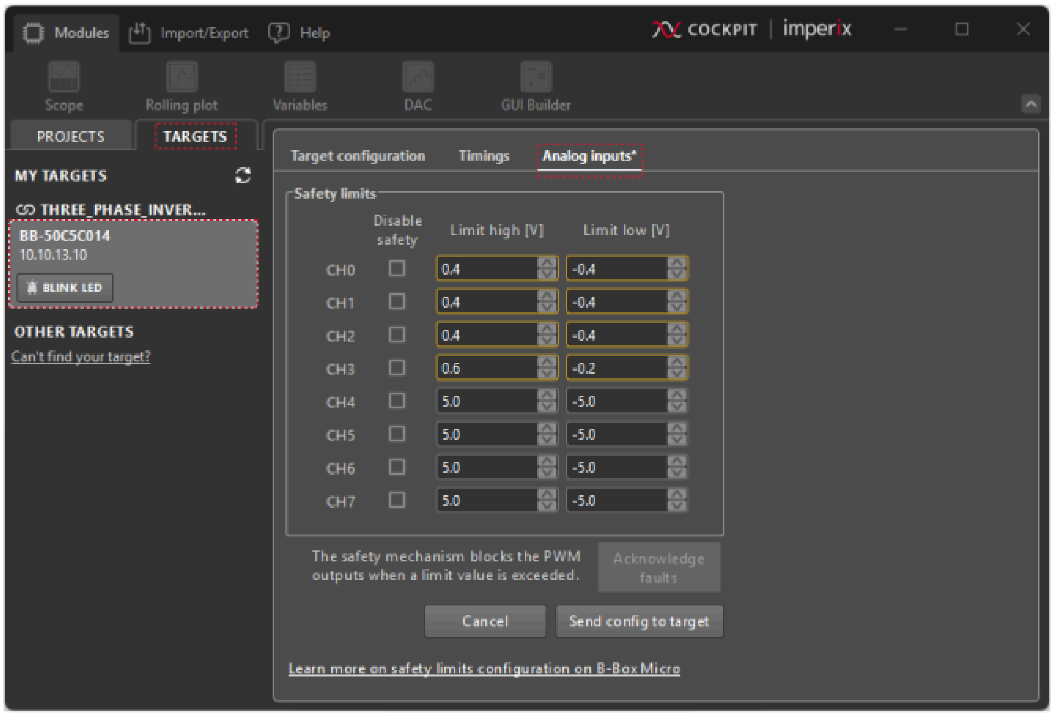
## How to configure analog inputs and protections

The B-Box Micro features safety limits, which can be configured on Cockpit. For that, when connected to a B-Box Micro, the *Target configuration* window offers an additional tab called Analog inputs (see illustration below). More details about analog inputs configuration on B-Box Micro are given in PN106.

### Safety limits configuration for the open loop three-phase inverter

Assuming that the load current is measured using the embedded current sensors of PEB8038 modules, whose sensitivity is 50.0 mV/A, the *Limit high* and *Limit low* values can be computed as follows. The same approach applies to the DC bus voltage sensor.

- Current sensors (CH0, CH1, CH2)
  - Maximum acceptable instantaneous current: ±8 A
  - Corresponding sensor output: ±8 A x 50.0 mV/A = 400 mV
  - Limit high: +0.4 V and limit low: -0.4 V
- Voltage sensor (CH3)
  - Maximum acceptable DC voltage: 120 V
  - Corresponding sensor output: 120 V x 4.99 mV/V ≈ 600 mV
  - Limit high: +0.6 V and limit low: -0.2 V



Safety limits

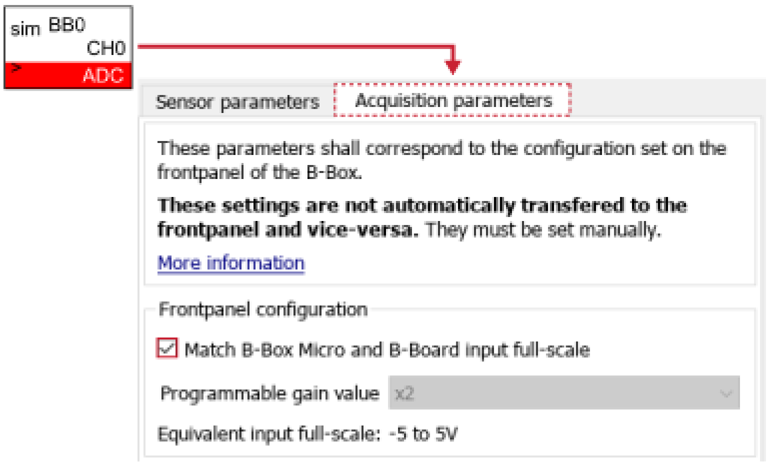
| | Disable safety | Limit high [V] | Limit low [V] |
|---|---|---|---|
| CH0 | ☐ | 0.4 | -0.4 |
| CH1 | ☐ | 0.4 | -0.4 |
| CH2 | ☐ | 0.4 | -0.4 |
| CH3 | ☐ | 0.6 | -0.2 |
| CH4 | ☐ | 5.0 | -5.0 |
| CH5 | ☐ | 5.0 | -5.0 |
| CH6 | ☐ | 5.0 | -5.0 |
| CH7 | ☐ | 5.0 | -5.0 |

The safety mechanism blocks the PWM outputs when a limit value is exceeded.

Learn more on safety limits configuration on B-Box Micro

Analog inputs configuration on B-Box Micro

On the software side, the ADC blocks must be configured with the suitable sensor parameters (sensitivity and offset). Finally, the option “Match B-Box Micro and B-Board input full-scale” in *Acquisition parameters* must be ticked.



These parameters shall correspond to the configuration set on the frontpanel of the B-Box.

**These settings are not automatically transfered to the frontpanel and vice-versa.** They must be set manually.

More information

Frontpanel configuration

☑ Match B-Box Micro and B-Board input full-scale

Programmable gain value: x2

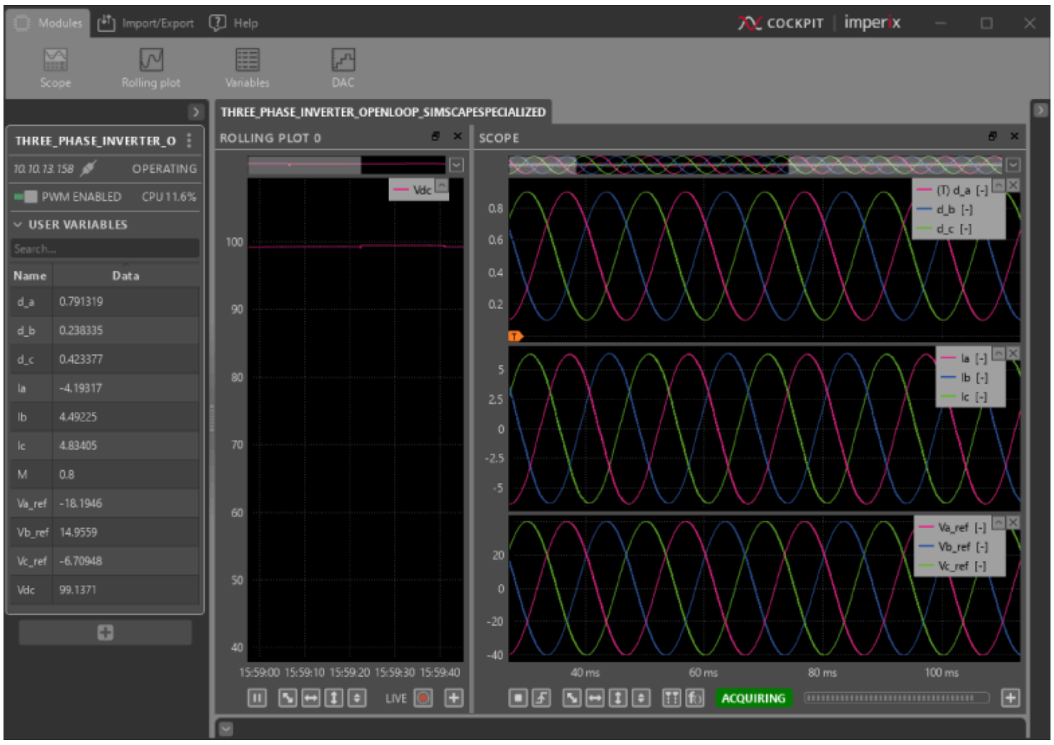
Equivalent input full-scale: -5 to 5V

How to match the scale on Simulink

## Run-time monitoring using Cockpit

Each Cockpit project contains a view in which users can drag and drop modules to monitor and alter user-defined variables. Any signal in the user code connected to a Probe block can be monitored in real time using the *Scope* or *Rolling plot* modules. Additionally, the Tunable parameters defined in the user code can be modified at run time from within the *Variables* module. A full guide on Cockpit monitoring software is addressed here.

Once the safety limits are configured, the PWM outputs can be enabled to monitor the behavior of the user-defined variables, as shown below.


THREE_PHASE_INVERTER_O

10.10.13.158 OPERATING

PWM ENABLED CPU 11.6%

USER VARIABLES

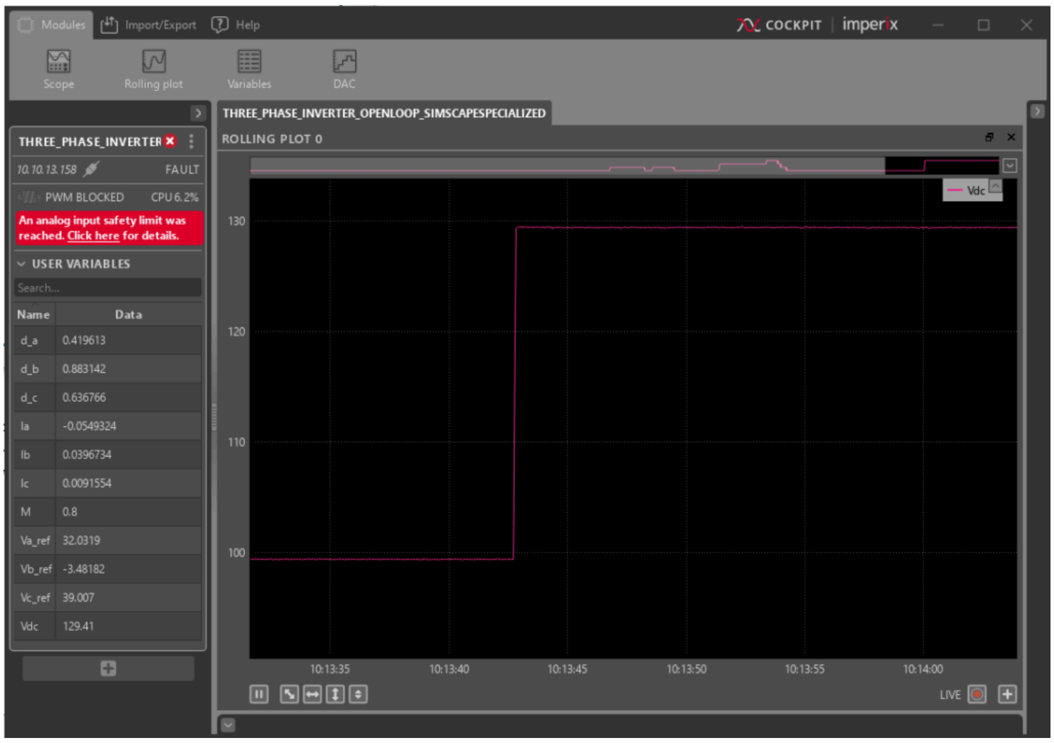
| Name | Data |
|---|---|
| d_a | 0.791319 |
| d_b | 0.238335 |
| d_c | 0.423377 |
| Ia | -4.19317 |
| Ib | 4.49225 |
| Ic | 4.83405 |
| M | 0.8 |
| Va_ref | -18.1946 |
| Vb_ref | 14.9559 |
| Vc_ref | -6.70948 |
| Vdc | 99.1371 |

Cockpit software – Providing run-time monitoring and parameter tuning

## How to acknowledge faults

If an error occurs during operation, the PWM outputs are blocked. To enable the PWM outputs again, the fault must be acknowledged.

For instance, let's consider a step change from 100 V to 130 V for the DC voltage supply. Since the high limit is set to 120 V, the voltage crosses the imposed threshold once the voltage step occurs. Hence, the PWM outputs are blocked, and the error message appears on the screen.



THREE_PHASE_INVERTER

10.10.13.158 FAULT

PWM BLOCKED CPU 6.2%

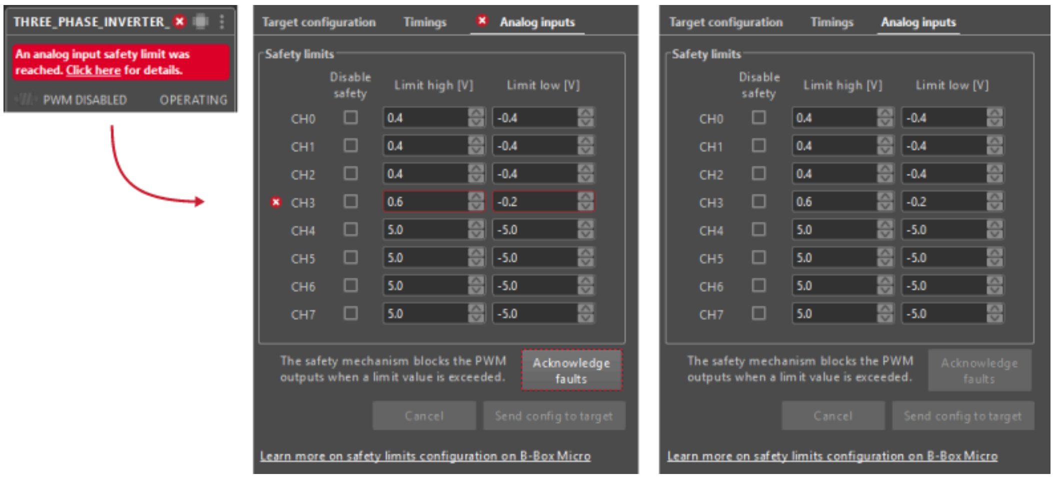
An analog input safety limit was reached. Click here for details.

USER VARIABLES

| Name | Data |
|---|---|
| d_a | 0.419613 |
| d_b | 0.883142 |
| d_c | 0.636766 |
| Ia | -0.0549324 |
| Ib | 0.0396734 |
| Ic | 0.0091554 |
| M | 0.8 |
| Va_ref | 32.0319 |
| Vb_ref | -3.48182 |
| Vc_ref | 39.007 |
| Vdc | 129.41 |

The Cockpit displaying an analog input safety limit threshold exceeded

To resume operation, the fault needs to be acknowledged. To do so, a shortcut on the error message is given to rapidly access the target limit configuration panel. The channel and the relative exceeded limit are highlighted in red. The button “Acknowledge faults” authorizes the return in the *operating* mode.



THREE_PHASE_INVERTER_

An analog input safety limit was reached. Click here for details.

PWM DISABLED OPERATING

Safety limits

| | Disable safety | Limit high [V] | Limit low [V] |
|---|---|---|---|
| CH0 | ☐ | 0.4 | -0.4 |
| CH1 | ☐ | 0.4 | -0.4 |
| CH2 | ☐ | 0.4 | -0.4 |
| CH3 | ☐ | 0.6 | -0.2 |
| CH4 | ☐ | 5.0 | -5.0 |
| CH5 | ☐ | 5.0 | -5.0 |
| CH6 | ☐ | 5.0 | -5.0 |
| CH7 | ☐ | 5.0 | -5.0 |

The safety mechanism blocks the PWM outputs when a limit value is exceeded.

Learn more on safety limits configuration on B-Box Micro

How to visualize which limit is triggered and acknowledge the fault on Cockpit

Considering the threshold limits imposed in Section 2.3, an error message for the current channels can be triggered by raising the value of the tunable parameter “M” (M stands for modulation index) from M=0.8 to M=1.

## Where to go next?

A wide spectrum of control techniques can be easily implemented and tested with the starter kit. The examples provided below are ready-to-use and fully tested:

- PI control:
  - PI current control
  - Vector current control
  - Cascaded voltage control
- Phase detectors:
  - Synchronous reference frame PLL
  - SOGI PLL
- Modulation techniques:
  - Carrier-based PWM
  - Space Vector PWM
  - Discontinuous PWM

- Advanced control techniques:
  - Proportional resonant controller
  - Maximum Power Point Tracking
  - Active damping of LCL filters
  - Model Predictive Control

Thanks to the modular nature of the modules and plug-and-play connectivity of the whole starter kit, many different applications and topologies can be addressed quickly and with high performance:

- Choppers:
  - Dual-Active bridge
  - Step-down buck converter
  - Step-up boost converter
  - Buck/Boost converter

- Inverters:
  - Single-phase inverter
  - Three-phase inverter
  - Three-phase grid-tie PV inverter

These are just a few examples of what is achievable with the 4x phase leg modules capabilities of the B-Box Micro.

Please note that B-Box Micro doesn't support distributed control. When multiple converters must be coordinated, for instance, for distributed traction or multilevel converters, control hierarchization is essential. For such advanced applications, the B-Box RCP should be preferred.