



# OpenECU Compatibility with Third Party Tools

Release 2.10.0 (r2024-1)



---

# Table of Contents

1. Third party tools .....	1
1.1. Third party tool requirements — C-API .....	1
1.2. Third party tool requirements — Simulink-API .....	1
1.3. Third party tool requirements — installation .....	2
1.4. Third party tool requirements — compatibility .....	2
1.5. Deprecated Tools .....	5

---

# Chapter 1. Third party tools

1.1. Third party tool requirements — C-API .....	1
1.2. Third party tool requirements — Simulink-API .....	1
1.3. Third party tool requirements — installation .....	2
1.4. Third party tool requirements — compatibility .....	2
1.5. Deprecated Tools .....	5

## 1.1. Third party tool requirements — C-API

For C based development, OpenECU requires (at a minimum) one of the following compiler tools:

- Wind River Diab compiler
- GCC Compiler

### Note

GCC is an optional component in the OpenECU installation (installed by default). Additionally, GCC support is currently in a beta stage. As such, there a number of known limitations for compiling an OpenECU application with GCC. Please see the “Integration notes for third party tools” of the “Release notes” for a list of known issues building with GCC for further details.

To program and calibrate an OpenECU with an application, OpenECU integrates with the following calibration tools. Only one calibration tool is required:

- OpenECU Calibrator
- ATI VISION
- Vector CANape

## 1.2. Third party tool requirements — Simulink-API

For Simulink model based development, OpenECU requires (at a minimum) the following MathWorks tools:

- MATLAB (base product)
- Simulink (to develop the models)
- Simulink Coder (to generate C code from the models)
- MATLAB Coder (Simulink Coder depends on this)

In addition, if you need to add state diagrams to the model, then you will also need:

- Stateflow (to develop state flow diagrams inside your model) Simulink Coder generates C code from the state flow diagrams inside your model.

Simulink Coder generates C code which does not lend itself to efficient repeatable testing. When creating a production version of your product, you may need better control of the

structure of the C code generated from the model to reduce the cost of testing the C code against any industry standards. Under these circumstances you will also need:

- Embedded Coder (to generate C code from the models)

To compile the generated C code (from either Simulink Coder or Embedded Coder), you will need one of the following compilers:

- Wind River Diab compiler
- GCC compiler (free compiler but with known issues)

To program and calibrate an OpenECU with an application, OpenECU integrates with the following calibration tools. Only one calibration tool is required:

- OpenECU Calibrator
- ATI VISION
- Vector CANape

## 1.3. Third party tool requirements — installation

OpenECU works with a number of applications (both required and optional) supplied by other companies. If you intend to use OpenECU with one of the following tools, it is best to install them before OpenECU. The installer will then integrate the OpenECU developer software with these applications. See [OpenECU Compatibility with Third Party Tools](#) for a list of supported versions.

- MATLAB
- ETAS INCA (deprecated) calibration tool

OpenECU works with a number of other applications, but these need not be installed prior to the OpenECU developer software.

- Simulink Coder, formerly Real-Time Workshop, (optional)
- Embedded Coder, formerly Real-Time Workshop Embedded Coder, (optional):
- Stateflow
- Wind River (Diab) C compiler
- OpenECU Calibrator
- ATI Vision calibration tool
- Vector CANape calibration tool

## 1.4. Third party tool requirements — compatibility

OpenECU has been tested against the latest versions of each tool listed below. OpenECU may work with other versions of these applications, but Pi only provides technical support for the latest version.

## Operating system

OpenECU works with the following versions of Windows.

Version	License	Installation and setup	Troubleshooting
Windows 10 Windows 11 <sup>a</sup>	Issued by Microsoft	Installation instructions provided by Microsoft. No special setup required.	No known issues

<sup>a</sup> OpenECU developer software may not function correctly on encrypted drives. OpenECU developer software must be able to create files on the host file system. If using an encrypted drive, be sure that permission settings will allow OpenECU to create files. Dana cannot provide support for issues with encrypted drives.

## Modeling tools

A modeling tool allows the user to diagrammatically describe their application logic and control. That tool generates source code which OpenECU automatically builds into an application using a compiler (next section).

OpenECU supports the following modeling versions of Mathworks MATLAB/Simulink

Version	License	Installation and setup	Troubleshooting
R2015b (deprecated) 32-bit	Issued by Mathworks	Installation instructions provided by Mathworks. Setup requires MATLAB's <code>PATH</code> variable to be adjusted, which the OpenECU installer can do for you, see OpenECU Developer Software Installation and Release Note.	No known issues
R2015b (deprecated) R2018a (deprecated) R2018b (deprecated) R2019a (deprecated) R2019b (deprecated) 64-bit <sup>a</sup>			
R2020a R2021b R2023a 64-bit			

<sup>a</sup> Deprecated versions of Matlab are supported in the current OpenECU release, but are planned to be removed in the next release.

## Compilers

A compiler translates C source code (either written by hand or generated by a modeling tool) into machine code that runs directly on the ECU.

All OpenECU targets use Freescale PowerPC microcontrollers. The M110, M220, M221, M250, M460 and M461 use an MPC5534 microcontroller, the M670 uses an MPC5674F microcontroller.

See the Technical Specification for your target for more information.

Compiler Version	Target ECU	License	Installation and setup	Troubleshooting
Diab v5.5.1.0 (deprecated) Diab v5.8.0.0 (deprecated)	M110, M220, M221, M250, M460, M461	Issued by Wind River	Installation instructions provided by Wind River. Setup requires the Window's <code>PATH</code> environment variable to be adjusted, or an	OpenECU Developer Software Installation and Release Note, "Diab 5.5.1.0, Known defects"
Diab v5.9.0.0 (deprecated) <sup>a</sup>	M110, M220, M221, M250,			none

Compiler Version	Target ECU	License	Installation and setup	Troubleshooting
Diab v5.9.6.7	M460, M461, M670		OpenECU specific environment variable to be created.	none
GCC v4.7.3 <sup>b</sup>	M110, M220, M221, M250, M460, M461, M670	none	Included with OpenECU installation	OpenECU Developer Software Installation and Release Note, "GCC 4.7.3, Known issues"

<sup>a</sup> Support for deprecated compiler versions will be removed in a future OpenECU release.

<sup>b</sup> OpenECU has only been tested using GCC Compiler version 4.7.3 and is in the beta stage. As such, there are a number of known issues to keep in mind when compiling an OpenECU application using GCC. For further details, please see "Integration notes for third party tools" for a list of known issues.

### Programming, Data Logging, and Calibration Tools

OpenECU requires a tool to program (or "Flash") the ECU with the application code from compilation. Once programmed, the ECU will execute the application. Interaction with the executing application requires a data logging or calibration tool to read and write information in the application.

These tools have been tested for reprogramming, data logging, and calibration capabilities. Some of them have many other features which have not been tested with OpenECU.

Tool	Version	License	Installation and setup	Troubleshooting
OpenECU Calibrator	Any	Issued by Dana	Installation instructions provided by Dana with the tool. No special setup required.	No known issues.
Vision	v2.5 through v6.0 <sup>a</sup>	Issued by ATI	Installation instructions provided by ATI.  The following Vision toolkits are typically used when working with OpenECU: <i>Data Acquisition Toolkit</i> , <i>Calibration Toolkit</i> , <i>Universal ECU Interface Standard Toolkit</i> , <i>APOLLO Data Analysis Toolkit</i> , <i>CAN Interface Toolkit</i> and <i>HORIZON Scripting/Remote API Toolkit</i> . In particular,	OpenECU Developer Software Installation and Release Note, "ATI Vision, Known defects"

Tool	Version	License	Installation and setup	Troubleshooting
			the <i>HORIZON Scripting/Remote API Toolkit</i> is required if OpenECU builds are to generate Vision strategy files (.vst).	
CANape	v8 through v17.0	Issued by Vector	Installation instructions provided by Vector. No special setup required.	No known issues.

<sup>a</sup> The OpenECU method of configuring ATI Vision uses standardised ASAP2 files. As a result, all future versions of Vision are expected to be backwardly compatible (e.g., version 3.7 and version 4.0 are known to be compatible).

## 1.5. Deprecated Tools

Some third party tools have been marked deprecated and support for these tools will be removed in a future release of OpenECU.

Third party tool	Replacement
MathWorks MATLAB R2015b	MATLAB latest version (see the <a href="https://openecu.com/product/openecu-third-party-tool-compatibility/">OpenECU website [https://openecu.com/product/openecu-third-party-tool-compatibility/]</a> for a complete list of supported versions of MATLAB).
MathWorks MATLAB R2018a	
MathWorks MATLAB R2018b	
MathWorks MATLAB R2019a	
MathWorks MATLAB R2019b	
ETAS INCA v7.2.7	Pisnoop    ATI Vision    Vector CANape
WindRiver Diab 5.5.1.0	WindRiver Diab 5.9.6.7
WindRiver Diab 5.8.0	WindRiver Diab 5.9.6.7
WindRiver Diab 5.9.0	WindRiver Diab 5.9.6.7