



EATHA20

Datasheet

V1.0.1

ECOTRON CORPORATION

www.ecotron.ai

Revision History

Revision Date	Version	Description
2025.12	V1.0.0	Initial Release

Contact us:

Web: <http://www.ecotron.ai>

Email: info@ecotron.ai

support@ecotron.ai

Address: 2942 Columbia Street

Torrance, CA, 90503

United States

Tel: +1 562-758-3039

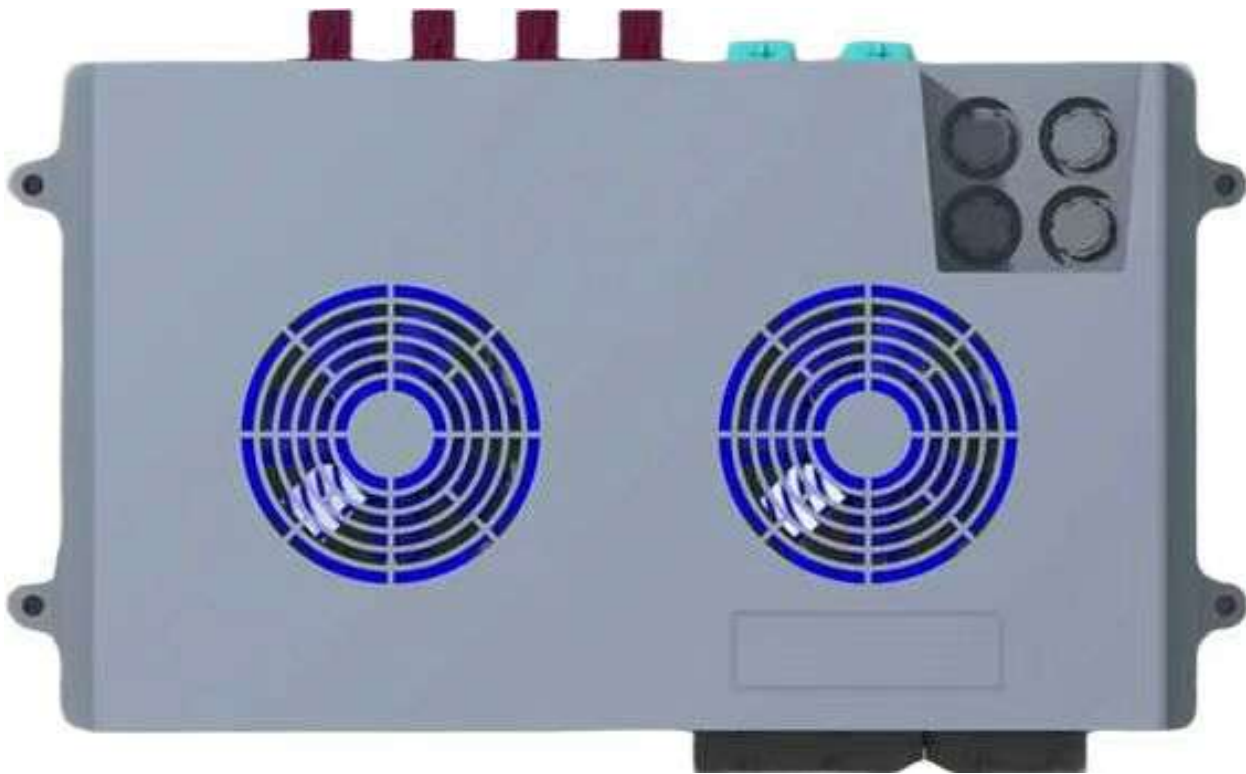
+1 562-713-1105

Table of Contents

1. OVERVIEW	4
2. INTERFACE CONFIGURATION	5
3. MECHANICAL STRUCTURE	7
3.1 DIMENSIONAL DRAWINGS	7
3.2 CONNECTORS	7
4. HARDWARE DESCRIPTION	9
4.1 SPECIFICATIONS	9
5. BASE SOFTWARE	10
6. APPLICATION SCENARIO	12
7. DEVELOPMENT TOOLS	13
7.1 ECOCODER	13
7.2 ECOCAL	15
7.3 ECOFLASH	15
8. INSTALLATION GUIDELINES	16

1. Overview

EATHA20 is designed with NVIDIA Jetson Thor + Infineon TC4D9 architecture, with a computing power of up to 2070TFLOPS, and is suitable for L3~L5 unmanned driving applications, which can integrate computing-intensive sensor data processing, fusion and control strategy development into a control unit. It can be used in unmanned logistics distribution, unmanned sanitation, unmanned mining vehicles, unmanned buses, intelligent construction machinery, and other applications with large computing power requirements.



2. Interface Configuration

Interface type	Quantity	Function	Internal Chip	Connector
Camera Interface	16	GMSL/GMSL2	SOC	4*4 cavity connector
Video Output Interface	2	FPDLINK	SOC	HSD connectors
Gigabit Automotive Ethernet	8	1000Base-T1	Switch	2*4 cavity connector
Standard Gigabit Ethernet	4	100BASE-T/1000BASE-T Standard	Switch	4* Aviation plug
10 Gigabit Optical Port	1	10G	SOC	SFP optical port
HDMI	1		SOC	
USB	1	One USB Host supports USB 2.0, USB 3.0, and USB 3.1.	SOC	
M.2 KEY M	2	System disk/extended storage	SOC	Inside
GNSS+IMU	1	Optional	SOC	
WIFI	1	Optional	SOC	
4G	1	Optional	SOC	
5G	1	Optional	SOC	
Audio interface	2	Optional	SOC	
CANFD	4		SOC	121PIN-CMC
RS232	3	1 port used for debug	SOC	
PPS_IN	1	Support 5V-16V	SOC	
PPM	1		SOC	
CAN	1		GNSS	
RS232	1		GNSS	
CANFD	8	2 channels with specific frame wake-up	MCU	
USS	12		MCU	
KEYON	3	1 for SOC and 2 for MCU	MCU	
Digital acquisition	6	Default configuration, 4-way high effective 2 ways low effective	MCU	

Analog acquisition	6		MCU	
Low-side drive	8	8 @250mA	MCU	
High-side drive	4	4 @1A	MCU	
5V sensor power supply	2	The maximum current is 100mA	MCU	
Positive power supply	6			
Ground power supply	7			
Signal ground	5			

3. Mechanical Structure

3.1 Dimensional drawings

The outside of the controller housing has no special treatment or plating, and there are no sharp burrs or sharp edges.

3.2 Connectors

The connector products used in the EATHA20 are qualified products that meet the safety level of automobiles, and the connector models are as follows:

Serial number	Connectors	Name	Category	Suppliers	Link
1	121P	PCB headers	1746979-1	TE	--
2		81P sheath	1473244-1	TE	http://www.digikey.com/products/en?keywords=1473244-1
3		40P sheath	1473252-1	TE	http://www.digikey.com/products/en?keywords=1473252-1
4		Large terminals	964273-2	TE	http://www.digikey.com/products/en?keywords=964273-2%20
5		Small terminals	968220-1	TE	http://www.digikey.com/products/en?keywords=968220-1
6		81P Back Cover	1473247-1	TE	http://www.digikey.com/products/en?keywords=1473247-1
7		40P back cover	1473255-1	TE	http://www.digikey.com/products/en?keywords=1473255-1
8		81P snap	368382-1	TE	http://www.digikey.com/products/en?keywords=368382-1
9		40P snap	368388-1	TE	http://www.digikey.com/products/en?keywords=368388-1
10	4-cavity waterproof	Plate end	2404815-1	TE	
11		Line end	<u>2-2354439-</u>	TE	

	connector		<u>1</u>		
12	4 cavity waterproof connectors	Plate end	E3SW4A-BMR131-S00	Amphenol	
13		Line end	E3WS4Z-WFS101-T02	Amphenol	
14	Aviation plug	Plate end	EEG.1K.308.CLN	ZRJM	
15		Line end	FGG.1K.308.CLA C	ZRJM	
16		Plate end	M8-F1-S8	DAOSM	
17		Line end	M8-D-P8	DAOSM	
18	Antenna interface	Waterproof antenna interface	SMA-KKY-22.2MM	YINSAIGE	

4. Hardware Description

The hardware circuit of the computing platform is designed according to the application needs of the autonomous driving system. The electrical parameters meet the requirements of the vehicle specification level and have a variety of data transmission interfaces to meet the needs of multi-sensor fusion of the autonomous driving system.

4.1 Specifications

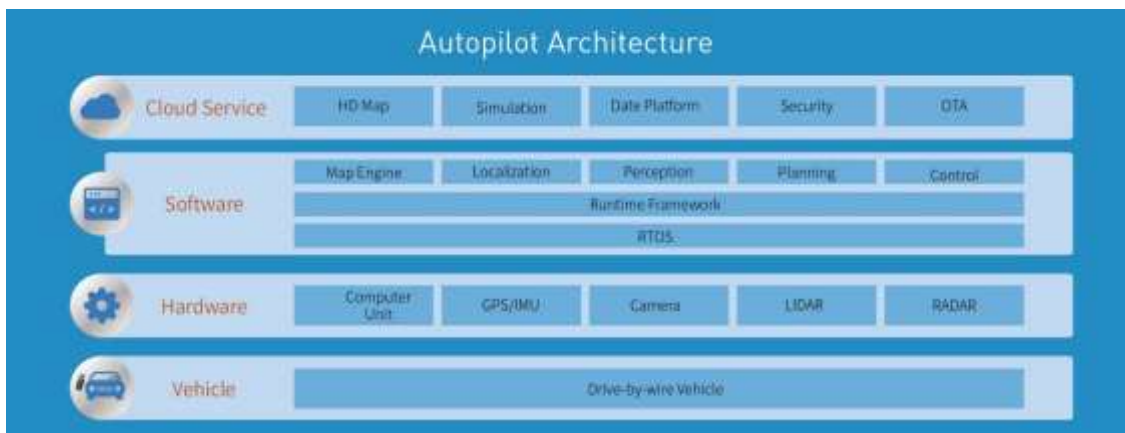
Project	Design Metrics
Operating voltage	DC 9~32V
Running space	128GB
Storage space	Refer to the size of the optional SSD
Operating temperature	-25~70°C
Operating humidity	0~95%, non-condensing
Storage temperature	-40~85°C
Waterproof	IP5X
Heat dissipation	Wind/water cooling for heat dissipation

5. Base Software

The SoC software system of this computing platform is custom developed for autonomous driving systems. The figure below shows a typical block diagram of an autonomous driving system. The Software system of this computing platform includes components such as the RTOS and the Runtime Framework. The RTOS is the Linux operating system, and the Runtime Framework is the melodic version of ROS (Robot Operating System).

The Linux operating system sits between the underlying hardware and the user, serving as a bridge between the two. Users can input commands through the operating system's user interface. The operating system interprets these commands, drives the hardware devices, and fulfills the user's requests. It provides the following functions: Processing management, Memory management, File system, Networking, Security, User interface, Device drivers, etc.

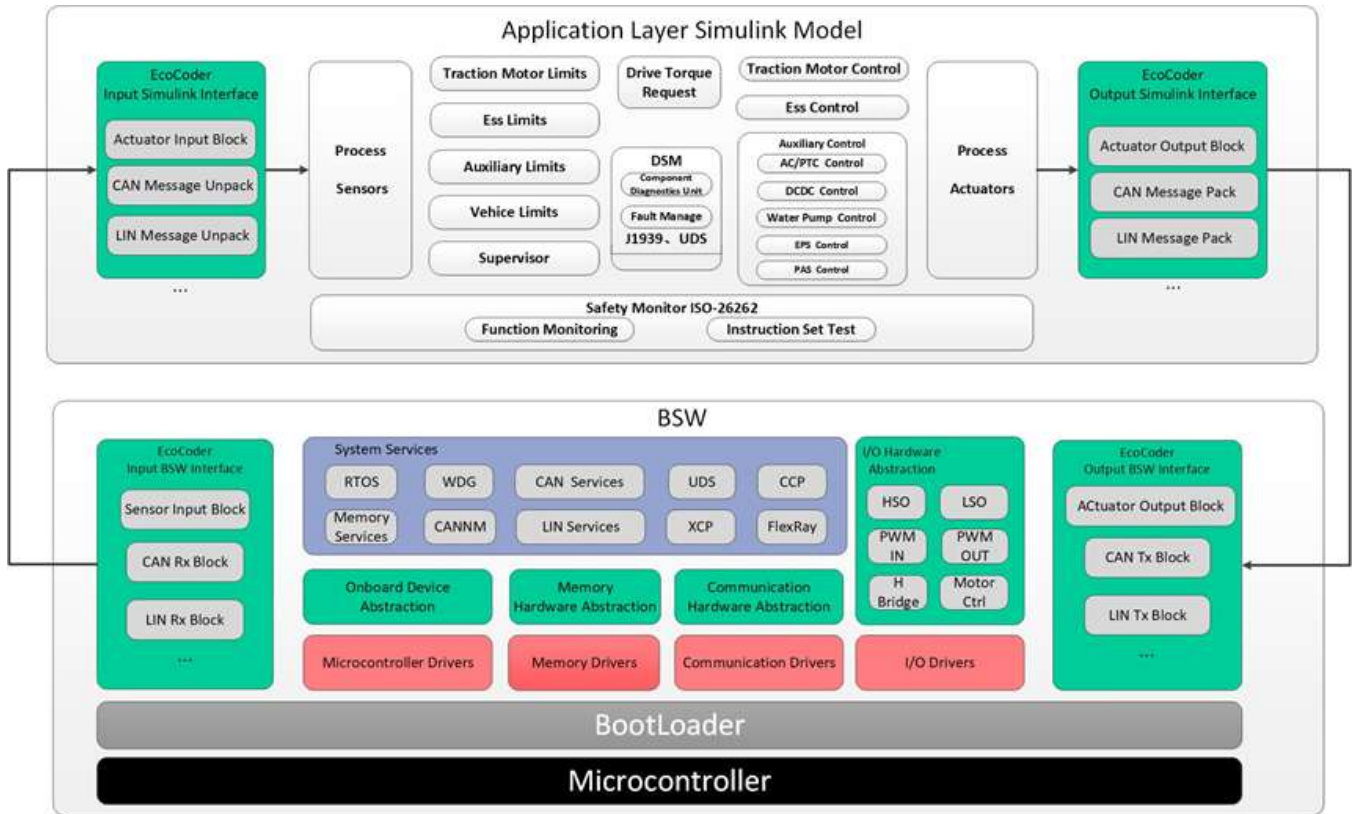
ROS provides a set of standard operating system services, such as hardware abstraction, low-level device control, implementation of commonly used functionalities, inter-process messaging, and data packet management. ROS is built on a graph-based architecture, enabling processes to run in different nodes to receive, publish and aggregate various types of information (such as sensing, control, status, planning, etc.).



The MCU software architecture within the Thor computing platform is designed in accordance with the AUTOSAR architecture standard. It is divided into the Application Software Layer and the Basic Software Layer. The Basic Software Layer is further divided into the Microcontroller Abstraction Layer, ECU Abstraction Layer, Service Layer, and Complex Drivers. The Application Software Layer and the Basic Software Layer are integrated and connected through EcoCoder. EcoCoder utilizes Simulink S-functions to encapsulate low-level interfaces into a Simulink module library. Application developers can build application-layer models in Simulink and generate executable program files compatible with the TC4D9 with a single click through Simulink.

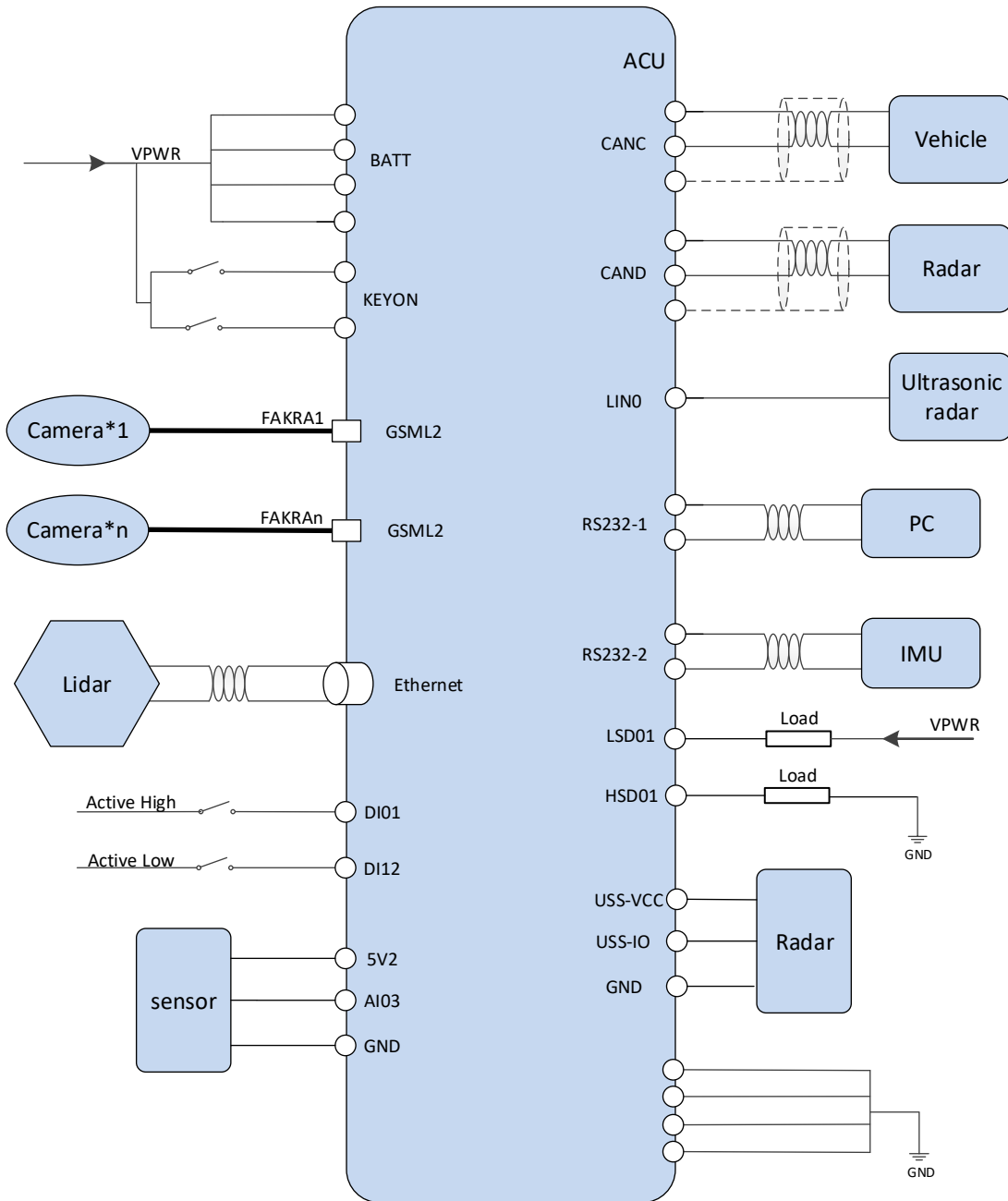
The low-level interfaces encapsulated by EcoCoder enable reading the status of digital input and analog signals, controlling high-side and low-side outputs, and supporting DBC file parsing as well as

protocols such as CCP and UDS. It also allows the definition of measurement variables, calibration parameters, and NVM variables. Together with the calibration software EcoCAL and the flashing software EcoFlash, it supports the development of MCU application programs.



6. Application Scenario

EATHA20 is used in the field of autonomous driving. It uses various sensors to form the hardware platform of an autonomous driving system. A common connection method is shown in the figure below:



7. Development Tools

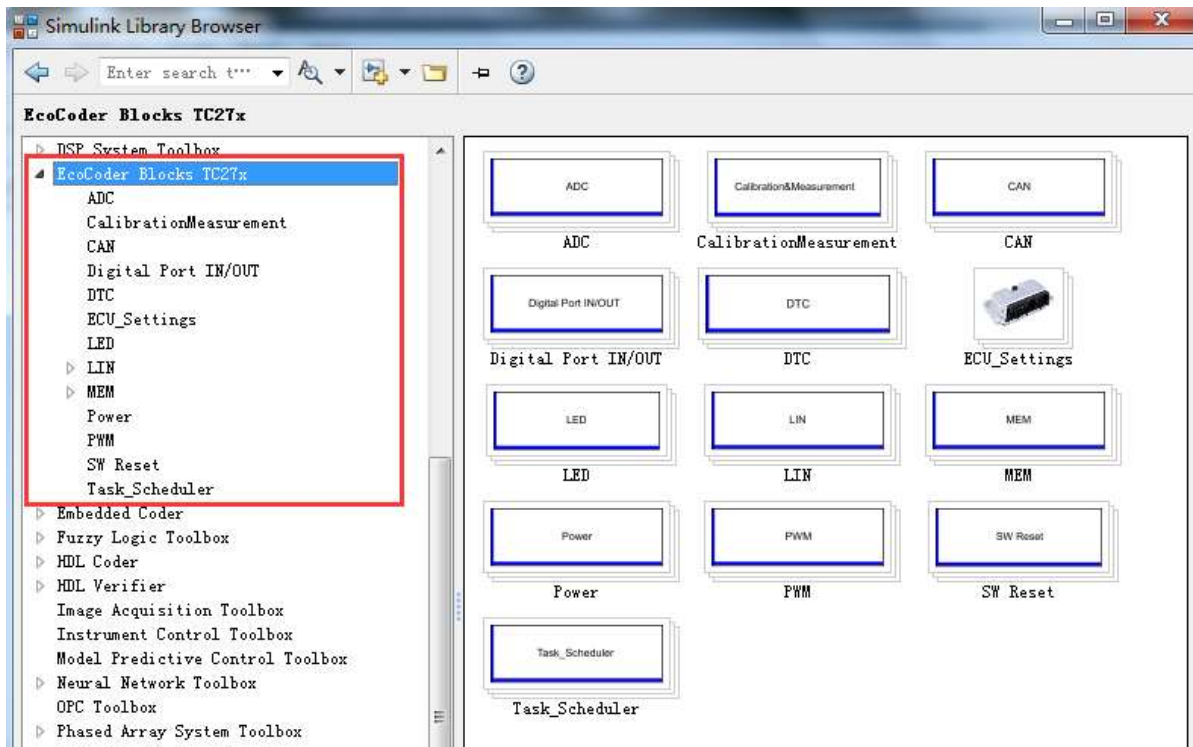
The MCU software architecture within the Thor computing platform is designed in accordance with the AUTOSAR architecture standard. It is divided into the Application Software Layer and the Basic Software Layer. The basic Software Layer is further subdivided into Microcontroller Abstraction Layer, ECU Abstraction Layer, Service Layer, and Complex Drivers Layer. The Application Software Layer and the Basic Software Layer are integrated and connected through the EcoCoder. EcoCoder uses Simulink S-functions to encapsulate low-level interfaces into a Simulink module library. Application developers can build application-layer models in Simulink and generate executable program files compatible with the TC4D9 with a single click in Simulink.

7.1 EcoCoder

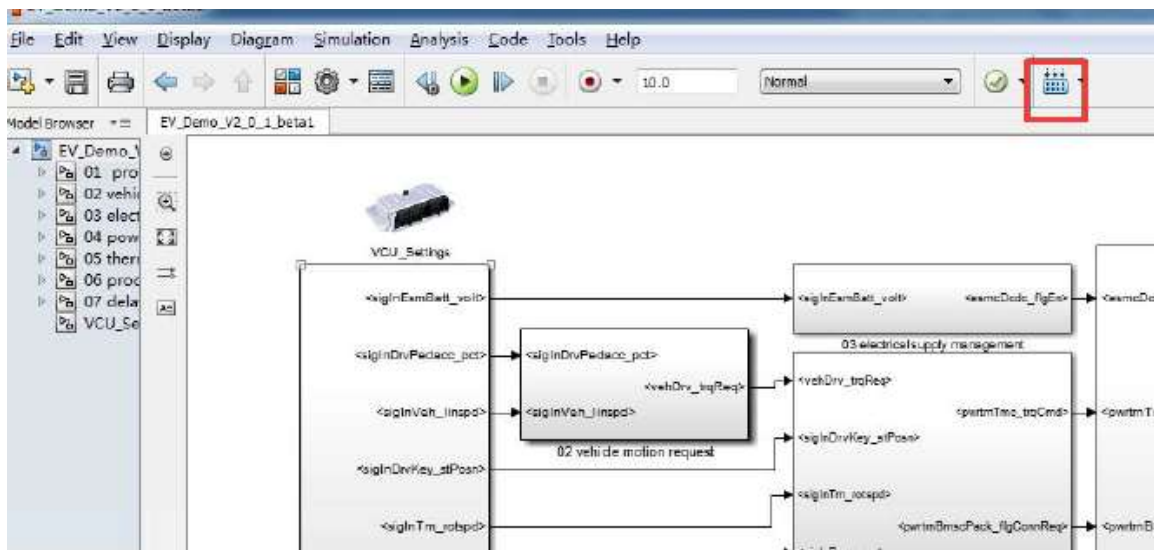
EcoCoder is an application development tool for control systems that enables users to more conveniently develop embedded application-layer programs within the Simulink environment. It extends Simulink and Real-Time Workshop Embedded Coder by providing the necessary code modules for generation and performs automatic configuration and optimization of the generated code. The basic software library is encapsulated as S-functions, allowing developers to use basic software interfaces in a graphical manner and perform basic parameter configuration. EcoCoder supports one-click generation of executable files and data description files and also provides an A2L address update tool.

Features:

- Users can develop embedded application-layer programs within the Simulink environment.
- Application-layer developers do not need to focus extensively on the microcontroller or hardware platform, allowing them to concentrate on control strategy development and reducing the involvement of cost of low-level software and hardware engineers.
- The basic software library is encapsulated as S-functions, enabling developers to use basic software interfaces graphically and perform parameter configurations, thereby lowering the difficulty for application-layer developers to use low-level software.
- Executable files and data description files can be generated from the application-layer model with a single click, and an A2L address update tool is provided.
- An automated integration approach is adopted, where the code generated from the application-layer model is automatically integrated with the low-level software in the background. The compiler is invoked automatically via a makefile to generate the executable file, eliminating the need for manual integration between the application and basic software layers and reducing the workload of the software integration engineers.



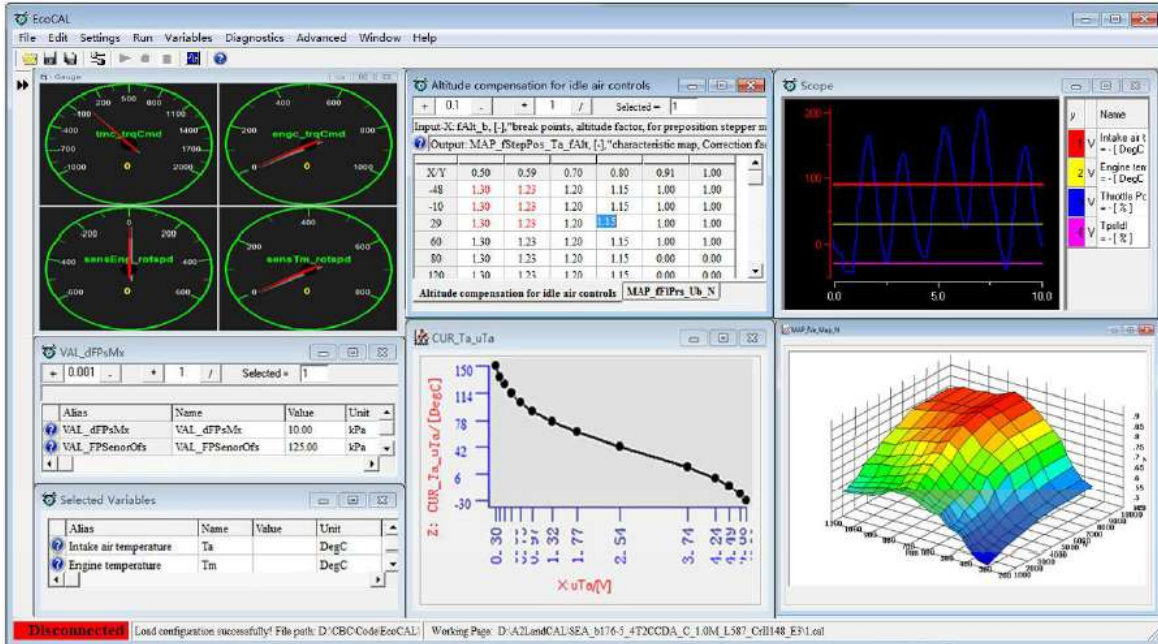
If the model simulation runs successfully, you can press the shortcut “Ctrl + B” or click the icon shown below to generate a flashable file with one click.



Application developers can use this tool to develop applications for the MCU inside the autonomous driving domain controller. For more details, please refer to the document “EcoCoder User Manual”.

7.2 EcoCAL

EcoCAL is a PC-based calibration software that operates using the CPP protocol. By loading A2L and Hex files, it enables real-time monitoring of measurement variables and online calibration of its calibration parameters. It assists control strategy development engineers in debugging and calibrating application software. For more details, please refer to the document “EcoCoder User Manual”.



7.3 EcoFlash

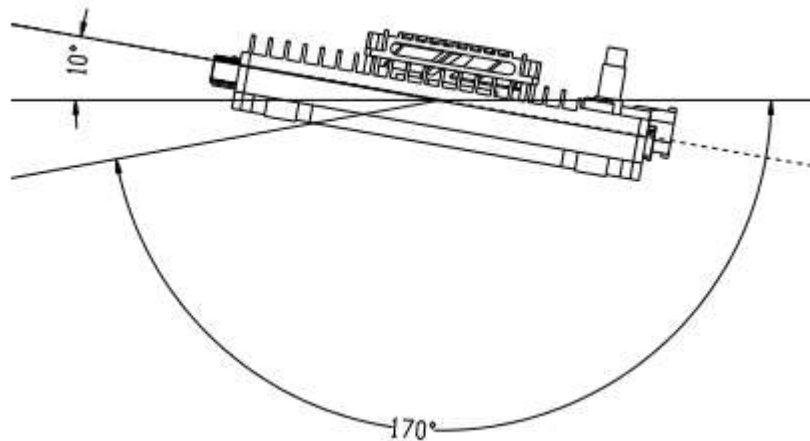
EcoFlash is our in-house developed PC software used in conjunction with the BootLoader to perform online flashing of program target files. It uses the CAN communication protocol with CPP/UDS support and can flash target files in .S19, .Mot, or .Hex formats.

8. Installation Guidelines

Ecotron recommends installing the ADCU inside of the user's vehicle cabin. If the OEM prefers to mount the ADCU in a different location, the installation position should be jointly evaluated by Ecotron's engineering team and engineers from the vehicle manufacturers.

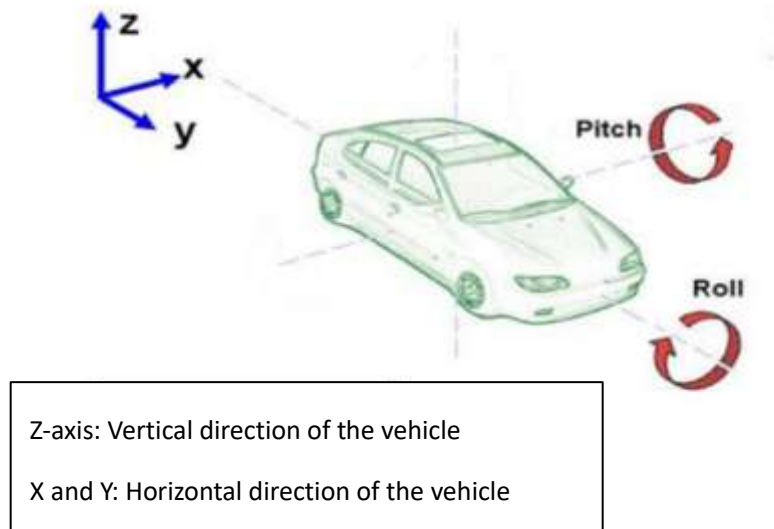
ADCU Installation Precautions:

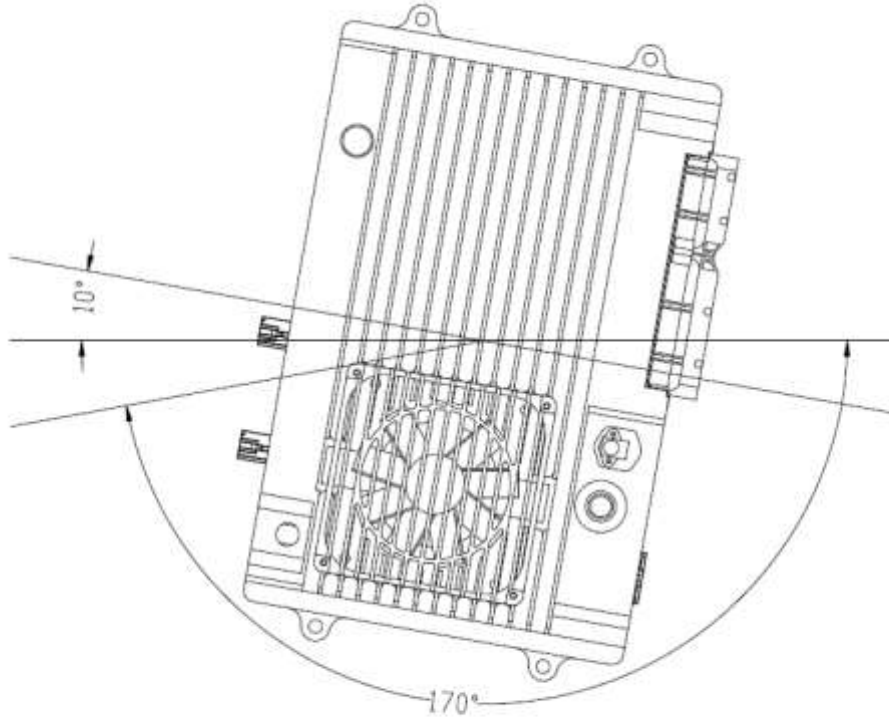
- 1) The ADCU and its wiring harness should be installed securely and reliably, with no looseness. The ADCU should not be used to support the harness. The harness layout must protect all wires from damage caused by abrasion or overheating.
- 2) Avoid installing the ADCU in areas where dust tends to accumulate, as excessive dust can affect its operational reliability.
- 3) The ADCU should be kept away from locations where its casing temperature could exceed the range of -15 to 70°C , and nearby components should not radiate excessive heat towards it. It is recommended to install the ADCU in areas with sufficient airflow to facilitate heat dissipation.



- 4) The ADCU should not be installed in locations prone to oil contamination, moisture, or water splashing.
- 5) The installation position and mounting method should prevent the ADCU from being subjected to additional mechanical vibrations or external impact. Avoid installing the ADCU at vehicle body resonance points.
- 6) Avoid installing the ADCU near areas where it could come into contact with the battery or other locations prone to leakage of acidic or alkaline solutions, as well as places where the ADCU may be exposed to corrosion.
- 7) The ADCU should not be installed near the battery's positive terminal or ignition power wiring terminals.

- 8) When installing the ADCU, the connectors should be angled downward both horizontally and vertically to prevent water from entering through the connectors. For horizontal installation, a range of -170° to -10° is recommended. For vertical installation, a range of -170° to -10° is recommended (See diagram below).





Ecotron recommends using the four built-in mounting points of the ADCU for installation and fixation. The mounting bracket is recommended to be made of metal, such as aluminum alloy, and the ADCU casing should have reliable electrical connection to the vehicle body through the bracket. If other material is used, the customer must ensure that they meet the ADCU's requirements for vibration, heat dissipation, temperature, EMC, and other factors. Any deviations should be confirmed with Ecotron's engineering team.

- 9) For Harness installation, high-speed signal lines such as network cables and video cables should, as much as possible, be routed away from high-voltage or high-interference areas such as motors, battery packs, and DC-DC converters.