

# Getting to know the VCG

The VCG (Vehicle Communication Gateway) allows users to bridge multiple modules and busses including CAN, CAN-FD, LIN and Automotive Ethernet with this single, innovative, easy to configure standalone data translation device. The VCG will host user-developed programs for applications such as node simulation, watchdog timer functions, CAN to CAN FD translation and more. Below, we'll explain how to get started working with the VCG.

## Configuration Software and Scripting Reference Guide

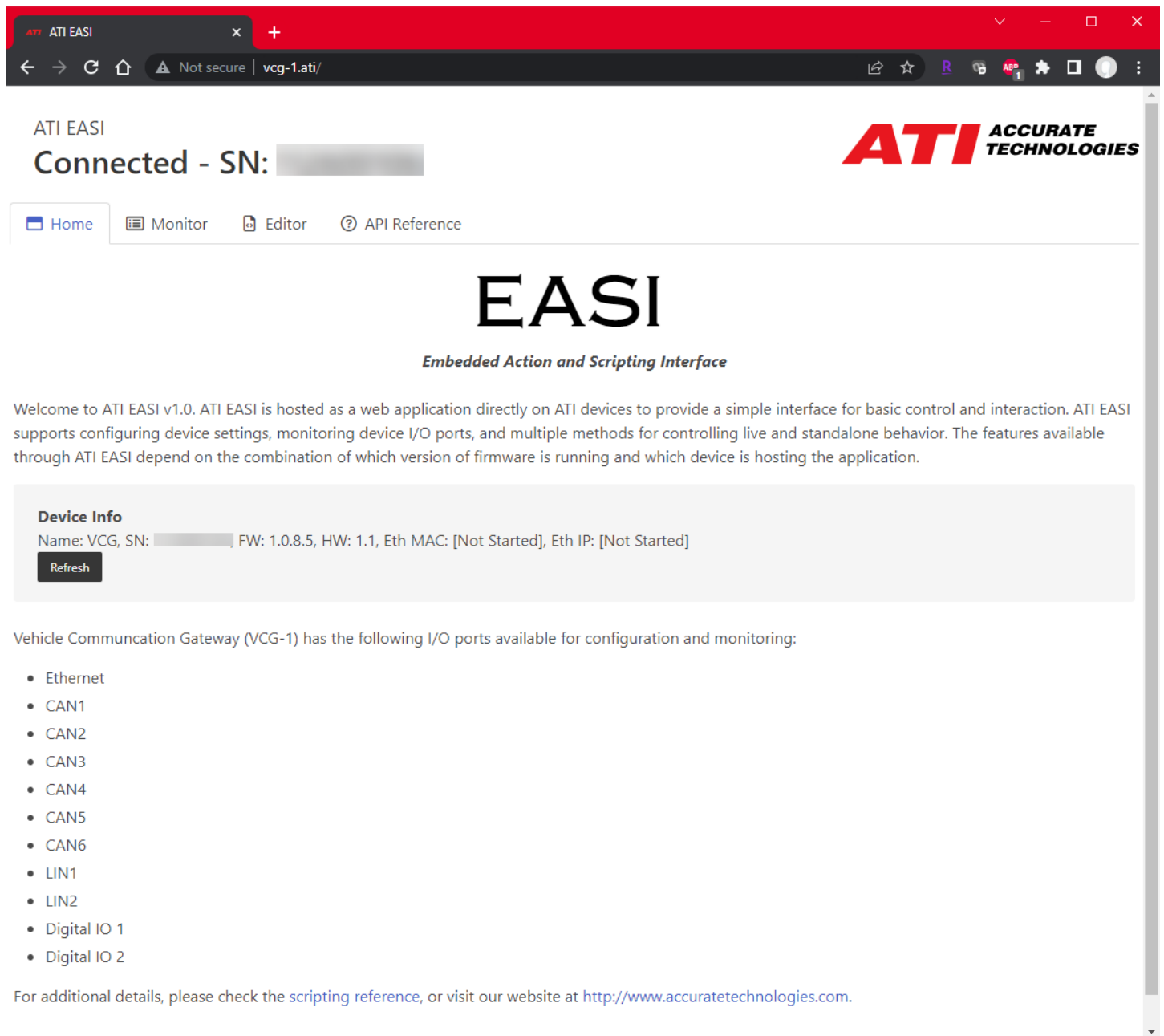
Unlike other VISION devices, the VCG has configuration software built into the unit allowing for configuration from any PC. When you first plug the VCG into your PC, it should detect the onboard memory card; recognized as a removable (USB) drive. The USB drive shows the contents of the memory card, you will see an executable named 'ATI EASI UI.exe.' This is the configuration software. Launching this file will open the VCG UI in a separate window. Also on the removable drive is a web link to the Scripting Reference Guide, named 'ATI EASI API Reference.html.' Launching this file will open in your default browser. No Internet connection is necessary.

Name	Date modified	Type	Size
.ATI	2/20/2023 10:20 AM	File folder	
ATI EASI API Reference.html	2/20/2023 10:20 AM	Chrome HTML Do...	184 KB
ATI EASI UI.exe	2/20/2023 10:20 AM	Application	504 KB
vcg-1.js	3/14/2023 2:49 PM	JavaScript File	2 KB



## ATI EASI WebUI

The ATI EASI WebUI will provides configuration utilities for the VCG including, a Monitor tab that allows you to see the traffic on the CAN and LIN channels, an Editor tab where you can edit, import/ export scripts, and save the configuration to the VCG, and a link to the VCG scripting API reference. The image below shows the Home tab which displays the name, serial number, Firmware level, hardware revision, and, if connected, the Ethernet MAC and IP address of your device. It also tells you what ports are available for configuration.



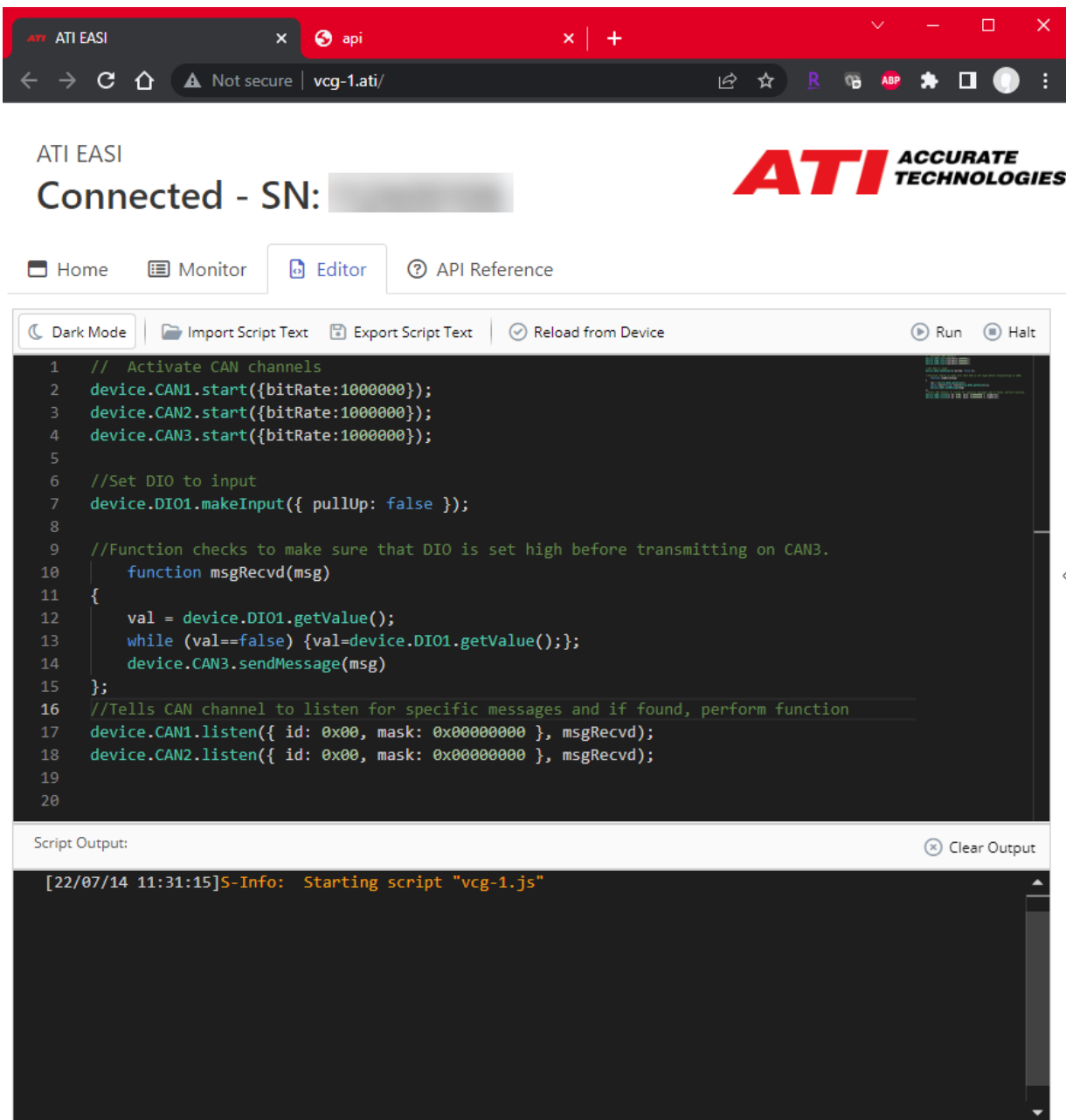
The screenshot shows a web browser window with the URL `vcg-1.ati/`. The page title is "ATI EASI" and the status is "Connected - SN: [REDACTED]". The ATI logo is in the top right. A navigation bar contains "Home", "Monitor", "Editor", and "API Reference". The main heading is "EASI" with the subtitle "Embedded Action and Scripting Interface". A welcome message states: "Welcome to ATI EASI v1.0. ATI EASI is hosted as a web application directly on ATI devices to provide a simple interface for basic control and interaction. ATI EASI supports configuring device settings, monitoring device I/O ports, and multiple methods for controlling live and standalone behavior. The features available through ATI EASI depend on the combination of which version of firmware is running and which device is hosting the application." Below this is a "Device Info" section with the text: "Name: VCG, SN: [REDACTED], FW: 1.0.8.5, HW: 1.1, Eth MAC: [Not Started], Eth IP: [Not Started]" and a "Refresh" button. A paragraph follows: "Vehicle Communication Gateway (VCG-1) has the following I/O ports available for configuration and monitoring:". A bulleted list of ports is shown: Ethernet, CAN1, CAN2, CAN3, CAN4, CAN5, CAN6, LIN1, LIN2, Digital IO 1, and Digital IO 2. At the bottom, a note says: "For additional details, please check the [scripting reference](#), or visit our website at <http://www accuratetechnologies.com>."

On the Monitor tab, you're able to select a CAN or LIN channel to monitor traffic on. Selecting the Start Monitor button on the right-hand side you can choose the Channel you wish to view, you can stop monitoring by selecting the Stop button, and you can save the monitor log to a .csv file by clicking on the Save log button. Note that the Monitor only works if you've started the channels you wish to view using the appropriate scripting commands in the editor.

The screenshot shows the ATI EASI web application interface. At the top, there is a navigation bar with tabs for Home, Monitor, Editor, and API Reference. The Monitor tab is active. Below the navigation bar, there is a header area with the text "ATI EASI Connected - SN: [redacted]" and the ATI logo. The main content area features a table with columns for #, Timestamp, Channel, Identifier, Flags, Length, and Data. A dropdown menu is open over the "Start Monitor" button, showing a list of channels: CAN Channel 1 (selected), CAN Channel 2, CAN Channel 3, CAN Channel 4, CAN Channel 5, CAN Channel 6, LIN Channel 1, and LIN Channel 2. The table contains several rows of data, all from CAN1.

#	Timestamp	Channel	Identifier	Flags	Length	Data
1347	1820.9440	CAN1	0x330		8	98 7
1348	1821.0270	CAN1	0x128		8	12 3
1349	1821.0440	CAN1	0x330		8	98 7
1350	1821.1270	CAN1	0x128		8	12 3
1351	1821.1460	CAN1	0x330		8	98 7
1352	1821.2270	CAN1	0x128		8	12 3
1353	1821.2440	CAN1	0x330		8	98 7
1354	1821.3270	CAN1	0x128		8	12 3

The Editor tab provides you with a scripting interface, complete with IntelliSense for device configuration. VCG scripts are coded in JavaScript using ECMA standards. The API reference includes the language extensions that are available for use with the VCG-1. You also have the ability to import and export your script text to .js (JavaScript) files if you wish to code in a different editing tool. The “Run” button at the top of the interface writes the script to the VCG, for when it’s run standalone, and allows you to start and test the VCG while it’s connected to your PC. Once you’ve written your script to the VCG, it’s ready for standalone operation. Every time the VCG is powered from the harness, the script will automatically run. The output window provides information when the script is run or halted, including if any syntax errors are detected. The code below shows an example of a simple gateway script that retransmits data from two different CAN channels to a single channel so long that the DIO1 input of the VCG is driven high.



ATI EASI  
Connected - SN: [REDACTED]

ATI ACCURATE TECHNOLOGIES

Home Monitor Editor API Reference

Dark Mode Import Script Text Export Script Text Reload from Device Run Halt

```
1 // Activate CAN channels
2 device.CAN1.start({bitRate:1000000});
3 device.CAN2.start({bitRate:1000000});
4 device.CAN3.start({bitRate:1000000});
5
6 //Set DIO to input
7 device.DIO1.makeInput({ pullUp: false });
8
9 //Function checks to make sure that DIO is set high before transmitting on CAN3.
10 function msgRecvd(msg)
11 {
12     val = device.DIO1.getValue();
13     while (val==false) {val=device.DIO1.getValue();};
14     device.CAN3.sendMessage(msg)
15 };
16 //Tells CAN channel to listen for specific messages and if found, perform function
17 device.CAN1.listen({ id: 0x00, mask: 0x00000000 }, msgRecvd);
18 device.CAN2.listen({ id: 0x00, mask: 0x00000000 }, msgRecvd);
19
20
```

Script Output: Clear Output

```
[22/07/14 11:31:15]S-Info: Starting script "vcg-1.js"
```

Combining CAN traffic from two different channels on to one, so long that DIO is driven high

```
// Activate CAN channels
device.CAN1.start({bitRate:1000000});
device.CAN2.start({bitRate:1000000});
device.CAN3.start({bitRate:1000000});

//Set DIO to input
device.DIO1.makeInput({ pullUp: false });

//Function checks to make sure that DIO is set high before transmitting on CAN3.
function msgRecvd(msg)
{
    val = device.DIO1.getValue();
    while (val==false) {val=device.DIO1.getValue();};
    device.CAN3.sendMessage(msg)
};
//Tells CAN channel to listen for specific messages and if found, perform function. All
messages on bus are listened for in this example.
device.CAN1.listen({ id: 0x00, mask: 0x00000000 }, msgRecvd);
device.CAN2.listen({ id: 0x00, mask: 0x00000000 }, msgRecvd);
```

## ATI VCG Scripting Reference Guide

The VCG Scripting Reference Guide can be accessed with the link on the SD card or from the API Reference tab in the WebUI. This provides all the commands you'll need to configure the VCG to suit your project. In addition, the guide provides example scripts for use cases like receiving and modifying CAN messages, receiving LIN messages and re-transmitting on CAN, and converting a multiplexed CAN message to multiple IDs.

**device**

The `device` object contains a set of objects which correspond to hardware components to allow interaction and direct control.

**device.CAN#**

The following objects represent the CAN buses available on the device. They all have the same functions and behavior:

- `device.CAN1`
- `device.CAN2`
- `device.CAN3`
- `device.CAN4`
- `device.CAN5`
- `device.CAN6`

**device.CAN1.start( settings )**

The `device.CAN1.start` function takes the CAN channel onto the bus with provided settings

**Parameter: settings**

The `settings` parameter is an object with the following properties:

Property Name	Required	Default	Description
<code>bitRate</code>	Yes	-	<i>[Value Type: number]</i> Defines the bitrate in bps for classic CAN and the

For additional questions, contact ATI support at [support@accuratetechnologies.com](mailto:support@accuratetechnologies.com).

Contact ATI Sales at : [sales@accuratetechnologies.com](mailto:sales@accuratetechnologies.com)

**US** +00 (1) 248 848 9200 / **China** +86 138 1023 6357 / **France** +33 (0) 1 72 76 26 10 / **Germany** +49 811 889 97351  
**India** +91 80 41255752 / **Japan** +81 3 6276 8950 / **Sweden** +46 (0) 31 773 7140 / **UK** +44 (0) 1767 652 340