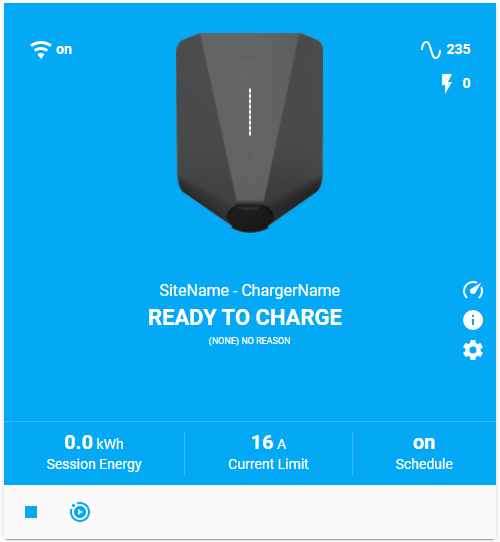
**EV Charger Card**

EV Charger card for [Home Assistant](https://www.home-assistant.io/) Lovelace UI

By default, Home Assistant does not provide any card for controlling chargers for electrical vehicles (EVs). This card displays the state and allows to control your charger - and it is now made fully customizable so you could also use it for «anything» else as well, for instance your electric car.

[](https://user-images.githubusercontent.com/54450177/97765425-56874900-1b12-11eb-9c0c-87721e0b5748.png)

**💡 Tip:** If you like this project consider buying me a cup of ☕️:

[](https://www.buymeacoffee.com/tmjo)

**Installation**

**Installation with HACS**

Installation with [HACS](https://hacs.xyz/) (Home Assistant Community Store) is highly reccomended but requires that you have this installed on your Home Assistant. The charger-card is available just by searching for Charger Card under the Frontend-section of HACS.

**Manual**

1. Download charger-card.js file from the [latest-release](https://github.com/tmjo/lovelace-charger-card/releases/latest).
2. Put charger-card.js file into your config/www folder.
3. Add reference to charger-card.js in Lovelace. There are two ways to do that:
   1. **Using UI:** *Configuration* → *Lovelace Dashboards* → *Resources* → Click Plus button → Set *Url* as /local/charger-card.js → Set *Resource type* as JavaScript Module.
   2. **Using YAML:** Add following code to lovelace section.
   3. resources:
   4. - url: /local/charger-card.js

type: module

1. Add custom:charger-card to Lovelace UI as any other card (using either UI-editor or YAML-configuration).

**Configuring and using the card**

After installation, the card can be configured either using Lovelace UI editor or YAML-code.

1. In Lovelace UI, click the 3 dots in top right corner and select *Edit Dashboard*
2. Click Plus button to add a new card.
3. Find *Custom: Charger Card* in the list.
4. Choose brand from the list which fits your need or select other.
5. Choose entity and select the main status sensor of your charger.
6. Now you should see the preview of the card!
7. Do your customizations in UI editor or manually in the YAML code editor.

Even if you prefer to configure an advanced and custom YAML-code, it is reccomended to use the UI at first since it will generate a template for the YAML-code that you can continue modifying.

Here is a list of the options. See [*advanced configuration*](https://github.com/tmjo/charger-card#advanced-configuration) further down for more details.

| **Name** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| type | string | **Required** | custom:charger-card |
| brand | string | **Required** | Select the template charger or system. If available it will try to set a pre-defined setup for your system and will assume you have all entities with default names from the integration. You can still edit your configuration in YAML if you system does not comply with these assumptions. See advanced config for further details. |
| entity | string | **Required** | The main status sensor for your charger (for Easee, look for the one named status). |
| customCardTheme | string | Optional | Select a built-in theme of colors, or use theme\_custom to apply the theme you have applied in HA. |
| chargerImage | string | Anthracite | Select a charger image from the built-in default images |
| customImage | string | Optional | Path to custom image of your charger (png or svg). This will override the chargerImage selection. (For images in the www-folder of HA, try path \local\image.png |
| compact\_view | boolean | false | Show a compact view of the card. |
| show\_name | boolean | true | Show a friendly name of the charger and the location (if provided) |
| show\_leds | boolean | true | Show status leds for the charger, fits best with Easee chargers and when using built-in images. |
| show\_status | boolean | true | Show status of the charger. |
| show\_collapsibles | boolean | true | Show the collapsible menu buttons which will open detailed sensors when clicked. |
| show\_toolbar | boolean | true | Show a toolbar with buttons that can call a service or do stuff. |
| show\_stats | boolean | true | Show data table (stats). |
| details | details | Optional | Details for customization if the default brand template doesn't do what you want it to. This mus be done manually with YAML-code. See [*advanced configuration*](https://github.com/tmjo/charger-card#advanced-configuration) |

**Features**

* Fully customizable for your needs, the card works for different chargers and other entities if you read the [*advanced configuration*](https://github.com/tmjo/charger-card#advanced-configuration) section and customize it to your needs. For chargers that are supported by default templates, the setup is very easy: select brand and main entity and you should be done.
* Animations: If choosing the default images of Easee chargers in any color, you can also choose to show leds which will behave according to charger status. This is identical to how the charger looks physically and similar to the Easee app and web site. Two leds for standby, all leds when connected, flashing while charging and so on. If SmartCharging is enabled, leds will be blue.
* Collapsible menu buttons: Click on one of the menu buttons (if you have enabled them) to get more info, config or limit settings. Can also be customized.
* Possibility to set current limits from UI. Current limits can also be customized.
* Customizable data table (stats) items that can depend on charger status and show relevant information.
* Customizable toolbar with buttons to call services or do something which. The toolbar can depend on charger status and show relevant actions.

**Supported languages**

This card supports translations. Please, help to add more translations and improve existing ones. Here's a list of supported languages:

* English
* Norsk bokmål (Norwegian)
* Svenska (by [jockesoft](https://github.com/jockesoft))
* German (by [DeerMaximum](https://github.com/DeerMaximum))
* Danish (by [dykandDK](https://github.com/dykandDK))
* Catalan (by [gerardag](https://github.com/gerardag))
* [*Your language?*](https://github.com/tmjo/lovelace-charger-card/tree/master/src/translations)

**Supported models**

This card was originally made to support charging robots from [Easee](https://easee-international.com/), but has been further developed to be fully configurable and customizable and can be used with any kind of charger. It can even be used for other things than EV-chargers, for instance your electric car or something completely different. Some brands are added with built-in support to make the configuration really easy:

* Easee
* [*Your charger?*](https://github.com/tmjo/lovelace-charger-card/edit/master/README.md)

**Advanced configuration**

If your brand is on the list of supported models, you should be able to get away just by selecting the brand and entity main sensor from the UI editor of Home Assistant. However, if you want to fully customize it or use it for something else than supported by default brands, you must do this yourself with YAML-code. It may seem overwhelming at first, so a trick can be applying a built-in brand as a template and start modifying the YAML code from there - this will give you the basic structure. Also see the examples further down.

| **Name** | **Type** | **Default** | **Description** |
| --- | --- | --- | --- |
| debug | boolean | false | Apply true to enable some debugging features which will appear in Developer Tools of your browser (Ctrl+Shift+I in Chrome). Useful if something doesn't work. |
| localize | boolean | true | Apply false to disable all translations, for instance if you want to fully customize texts and the translation feature for some reason messes up you stuff. Normally not necessary. |
| details | details |  | Details for customization if the default brand template doesn't do what you want it to. See [details-items](https://github.com/tmjo/charger-card#details-items) |

**Details items**

| **Config** | **Text** |
| --- | --- |
| name | Single entity with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). |
| location | Single entity with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). |
| status | Single entity with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). |
| substatus | Single entity with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). |
| smartcharging | Single entity with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). Controls white or blue leds for easee chargers. |
| info\_left | List of entities with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). Defines static icons shown top-left of the card. |
| info\_right | Same as info\_left, but for static icons shown top-right of the card. |
| group1 | List of entities with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). Defines the content of first collapsible-button (default: limits). |
| group2 | List of entities with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). Defines the content of second collapsible-button (default: info). |
| group3 | List of entities with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). Defines the content of third collapsible-button (default: config). |
| stats | Provide the states of entity defined as main status sensor, then a list of entities with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). Defines the datatable statistics (stats) in lower part of card. It may change depending on status of charger, or it is possible to define default as a state. See example. |
| toolbar\_left | Provide the states of entity defined as main status sensor, then a list of entities with configuration details as shown in [entity-items](https://github.com/tmjo/charger-card#entity-items). Defines the left aligned command buttons on the toolbar at the bottom of the card. It may change depending on status of charger, or it is possible to define default as a state. See example. |
| toolbar\_right | Same as toolbar\_left but right aligned command buttons on the toolbar. |
| currentlimits | Override the card default current limits by specifying a list of numbers. Used for current limit features of the card (if configured). You can also use dropdownitems for each item to customize the dropdown if you prefer. |
| statetext | Override or custom-translate the status sensor state by providing better or cleaner text. See example. You may force off translation by adding a leading '\_' to your text, but it is usually not neccessary. |
| collapsiblebuttons | Specify text and/or icon to customize the collapsible-buttons group1,group2 and group3 respectively. See example. |

**Entity items**

| **Config** | **Text** |
| --- | --- |
| entity\_id | Specify an entity id that you want to use for the specific feature. Enter calculated to perform mathematical functions instead, see further down how. |
| attribute | You can specify an attribute of the entity to use instead of entity state as value. |
| unit | Card will use the entity unit\_of\_measurement as default, but you can override it here by entering the unit you want. |
| unit\_show | Define if the unit should be shown next to the value or not. |
| unit\_showontext | Define if the value and unit should be shown next to the tooltip text |
| text | Card will use the entity friendly\_name as default, but you can override it here by entering the text you want. Used as tooltip text for icons and as information on stats. |
| service | You can specify a service that you wish to link to the entity, check available services in HA Developer Tools. Use full format like domain.servicename |
| service\_data | Specify the service data corresponding to the service you want to call. If it is not a static call, but for instance a call with a value selected from a dropdown menu, enter #SERVICEVAL# as a special token which will be replaced by the card when called. See advanced configuration for example. |
| icon | Card will use the entity icon as default, and will try to detect device class icons as well. If you want to override it, do it here by specifying a valid icon. |
| round | If you would like to round down the number, specify an integer to define the number of digits. |
| type | Choose between info, service and dropdown. Info-items will appear as icons with tooltip, clicking them opens the more-info popup for the entity. Service-items will do the same, but call a service when clicked (if it is provided). Dropdown-items will show a dropdown list to select from a list and call a specified service with this number as service data. |
| dropdownitems | List of items used for dropdown values if the type is dropdown. Defaults to currentlimits if not specified. |
| calc\_function | Choose between min, max, mean, sum to define a mathematical function to be performed on a specified list of entities or attributes. The result will be used on the sensor provided you specify calculated instead of a real entity\_id. Then add a list of entities to be calculated in calc\_entities. |
| calc\_entities | A list of entites or entity attributes that are used for the mathematical function. Specify a list of entity\_ids and add attributes if you prefer to use that. |
| conditional\_entity | Specify a boolean entity (returning on/off or true/false) to define if the entity should be shown based on state condition. |
| conditional\_attribute | Specify a boolean entity and its attribute (returning on/off or true/false) to define if the entity should be shown based on state attribute condition. |
| conditional\_invert | Invert the conditional rule, so that true/on means hide and false/off means show |

**Advanced example (Easee charger)**

type: custom:charger-card

entity: sensor.CHARGERNAME\_status

customCardTheme: theme\_custom

chargerImage: Red

brand: easee

show\_leds: true

details:

name:

entity\_id: sensor.CHARGERNAME\_status

attribute: name

location:

entity\_id: sensor.CHARGERNAME\_status

attribute: site\_name

status:

entity\_id: sensor.CHARGERNAME\_status

substatus:

entity\_id: sensor.CHARGERNAME\_reason\_for\_no\_current

smartcharging:

entity\_id: switch.CHARGERNAME\_smart\_charging

currentlimits:

- 0

- 6

- 10

- 16

- 20

- 25

- 32

statetext:

disconnected: disconnected

awaiting\_start: awaiting\_start

charging: charging

completed: completed

error: error

ready\_to\_charge: ready\_to\_charge

collapsiblebuttons:

group1:

text: click\_for\_group1

icon: mdi:speedometer

group2:

text: click\_for\_group2

icon: mdi:information

group3:

text: click\_for\_group3

icon: mdi:cog

info\_left:

- entity\_id: binary\_sensor.CHARGERNAME\_online

text: online

info\_right:

- entity\_id: sensor.CHARGERNAME\_voltage

text: voltage

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_power

text: power

unit\_show: true

group1:

- entity\_id: sensor.CHARGERNAME\_dynamic\_charger\_limit

text: dyn\_charger\_limit

service: easee.set\_charger\_dynamic\_limit

service\_data:

charger\_id: CHARGERID

current: '#SERVICEVAL#'

- entity\_id: sensor.CHARGERNAME\_dynamic\_circuit\_limit

text: dyn\_circuit\_limit

service: easee.set\_charger\_circuit\_dynamic\_limit

service\_data:

charger\_id: CHARGERID

currentP1: '#SERVICEVAL#'

- entity\_id: sensor.CHARGERNAME\_max\_charger\_limit

text: max\_charger\_limit

service: easee.set\_charger\_max\_limit

service\_data:

charger\_id: CHARGERID

current: '#SERVICEVAL#'

- entity\_id: sensor.CHARGERNAME\_max\_circuit\_limit

text: max\_circuit\_limit

service: easee.set\_circuit\_max\_limit

service\_data:

charger\_id: CHARGERID

currentP1: '#SERVICEVAL#'

- entity\_id: sensor.CHARGERNAME\_offline\_circuit\_limit

text: offline\_circuit\_limit

service: easee.set\_charger\_circuit\_offline\_limit

service\_data:

charger\_id: CHARGERID

currentP1: '#SERVICEVAL#'

group2:

- entity\_id: binary\_sensor.CHARGERNAME\_online

text: online

- entity\_id: sensor.CHARGERNAME\_voltage

text: voltage

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_power

text: power

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_current

text: charger\_current

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_circuit\_current

text: circuit\_current

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_energy\_per\_hour

text: energy\_per\_hour

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_session\_energy

text: session\_energy

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_lifetime\_energy

text: lifetime\_energy

unit\_show: true

group3:

- entity\_id: switch.CHARGERNAME\_is\_enabled

text: enabled

- entity\_id: switch.CHARGERNAME\_enable\_idle\_current

text: idle\_current

- entity\_id: binary\_sensor.CHARGERNAME\_cable\_locked

text: cable\_locked

- entity\_id: switch.CHARGERNAME\_cable\_locked\_permanently

text: perm\_cable\_locked

- entity\_id: switch.CHARGERNAME\_smart\_charging

text: smart\_charging

- entity\_id: sensor.CHARGERNAME\_cost\_per\_kwh

text: cost\_per\_kwh

- entity\_id: binary\_sensor.CHARGERNAME\_update\_available

text: update\_available

- entity\_id: binary\_sensor.CHARGERNAME\_basic\_schedule

text: schedule

stats:

default:

- entity\_id: sensor.CHARGERNAME\_session\_energy

text: session\_energy

unit\_show: true

- entity\_id: switch.CHARGERNAME\_cable\_locked\_permanently

text: cable\_locked

- entity\_id: binary\_sensor.CHARGERNAME\_basic\_schedule

text: schedule

disconnected:

- entity\_id: sensor.CHARGERNAME\_session\_energy

text: session\_energy

unit\_show: true

- entity\_id: switch.CHARGERNAME\_cable\_locked\_permanently

text: cable\_locked

- entity\_id: calculated

text: used\_limit

unit: A

unit\_show: true

calc\_function: min

calc\_entities:

- entity\_id: sensor.CHARGERNAME\_dynamic\_charger\_limit

- entity\_id: sensor.CHARGERNAME\_dynamic\_circuit\_limit

- entity\_id: sensor.CHARGERNAME\_max\_charger\_limit

- entity\_id: sensor.CHARGERNAME\_max\_circuit\_limit

- entity\_id: sensor.CHARGERNAME\_offline\_circuit\_limit

awaiting\_start:

- entity\_id: sensor.CHARGERNAME\_session\_energy

text: session\_energy

unit\_show: true

- entity\_id: binary\_sensor.CHARGERNAME\_basic\_schedule

text: schedule

- entity\_id: switch.CHARGERNAME\_smart\_charging

text: smart\_charging

- entity\_id: calculated

text: used\_limit

unit: A

unit\_show: true

calc\_function: min

calc\_entities:

- entity\_id: sensor.CHARGERNAME\_dynamic\_charger\_limit

- entity\_id: sensor.CHARGERNAME\_dynamic\_circuit\_limit

- entity\_id: sensor.CHARGERNAME\_max\_charger\_limit

- entity\_id: sensor.CHARGERNAME\_max\_circuit\_limit

- entity\_id: sensor.CHARGERNAME\_offline\_circuit\_limit

charging:

- entity\_id: sensor.CHARGERNAME\_session\_energy

text: session\_energy

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_energy\_per\_hour

text: energy\_per\_hour

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_circuit\_current

text: circuit\_current

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_output\_limit

text: output\_limit

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_current

text: current

unit\_show: true

- entity\_id: sensor.CHARGERNAME\_power

text: power

unit\_show: true

completed:

- entity\_id: sensor.CHARGERNAME\_session\_energy

text: session\_energy

unit\_show: true

- entity\_id: binary\_sensor.CHARGERNAME\_basic\_schedule

text: schedule

- entity\_id: calculated

text: used\_limit

unit: A

unit\_show: true

calc\_function: min

calc\_entities:

- entity\_id: sensor.CHARGERNAME\_dynamic\_charger\_limit

- entity\_id: sensor.CHARGERNAME\_dynamic\_circuit\_limit

- entity\_id: sensor.CHARGERNAME\_max\_charger\_limit

- entity\_id: sensor.CHARGERNAME\_max\_circuit\_limit

- entity\_id: sensor.CHARGERNAME\_offline\_circuit\_limit

error:

- entity\_id: sensor.CHARGERNAME\_session\_energy

text: session\_energy

unit\_show: true

- entity\_id: binary\_sensor.CHARGERNAME\_basic\_schedule

text: schedule

ready\_to\_charge:

- entity\_id: sensor.CHARGERNAME\_session\_energy

text: session\_energy

unit\_show: true

- entity\_id: binary\_sensor.CHARGERNAME\_basic\_schedule

text: schedule

- entity\_id: calculated

text: used\_limit

unit: A

unit\_show: true

calc\_function: min

calc\_entities:

- entity\_id: sensor.CHARGERNAME\_dynamic\_charger\_limit

- entity\_id: sensor.CHARGERNAME\_dynamic\_circuit\_limit

- entity\_id: sensor.CHARGERNAME\_max\_charger\_limit

- entity\_id: sensor.CHARGERNAME\_max\_circuit\_limit

- entity\_id: sensor.CHARGERNAME\_offline\_circuit\_limit

toolbar\_left:

default:

- {}

disconnected:

- {}

awaiting\_start:

- service: easee.stop

service\_data:

charger\_id: CHARGERID

text: stop

icon: hass:stop

- service: easee.resume

service\_data:

charger\_id: CHARGERID

text: resume

icon: hass:play

- service: easee.override\_schedule

service\_data:

charger\_id: CHARGERID

text: override

icon: hass:motion-play

charging:

- service: easee.stop

service\_data:

charger\_id: CHARGERID

text: stop

icon: hass:stop

- service: easee.pause

service\_data:

charger\_id: CHARGERID

text: pause

icon: hass:pause

completed:

- service: easee.stop

service\_data:

charger\_id: CHARGERID

text: stop

icon: hass:stop

- service: easee.override\_schedule

service\_data:

charger\_id: CHARGERID

text: override

icon: hass:motion-play

error:

- service: easee.reboot

service\_data:

charger\_id: CHARGERID

text: reboot

icon: hass:restart

ready\_to\_charge:

- service: easee.stop

service\_data:

charger\_id: CHARGERID

text: stop

icon: hass:stop

- service: easee.override\_schedule

service\_data:

charger\_id: CHARGERID

text: override

icon: hass:motion-play

toolbar\_right:

default:

- service: persistent\_notification.create

service\_data:

message: Firmware update is available, but only possible when disconnected!

title: Update

text: update

icon: mdi:file-download

conditional\_entity: binary\_sensor.CHARGERNAME\_update\_available

disconnected:

- service: easee.update\_firmware

service\_data:

charger\_id: CHARGERID

text: update

icon: mdi:file-download

conditional\_entity: binary\_sensor.CHARGERNAME\_update\_available

**Development**

Want to contribute to the project? Translations? Adding your brand as a pre-defined template?

First of all, thanks! Check [contributing guideline](https://github.com/tmjo/charger-card/blob/master/CONTRIBUTING.md) for more information.

**License**

MIT © [Tor Magne Johannessen](https://github.com/tmjo)

This project is heavily inspired by [denysdovhan](https://github.com/denysdovhan), and his [vacuum card](https://github.com/denysdovhan/vacuum-card). Thanks!