|  |
| --- |
| Ignition sense using LuvitRED |
|  |





**Franco Arboleda**

**25-Oct-16**

Table of Contents

1 Introduction 3

2 Setting up the GPIO for Ignition detection. 4

3 Wake up the CloudGate on ignition sense. 7

4 Timed Wake up 9

5 Configuration example 11

# Introduction

This document covers the configuration of ignition sense on the CloudGate using LuvitRED.

Ignition sense is a CloudGate feature currently only available on the CloudGate LTE (CG0114) using firmware version 2.x.x. This feature allows the user to power on the CloudGate by pulling high the ignition sense input line on the power connector.

The pinout of the power connector, seen from the terminal side, is as follows:

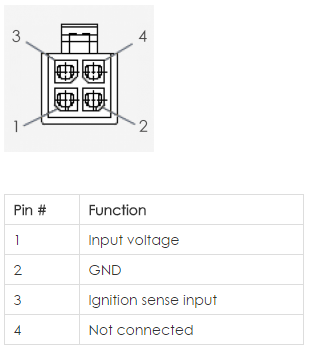
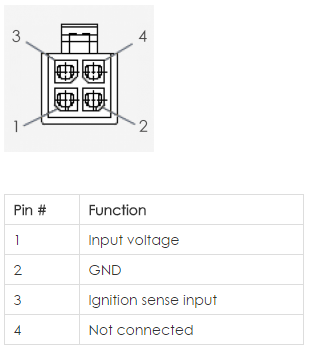


Figure : Power connector Pinout specification.

The electrical specification for the Ignition sense input is as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Min** | **Max** | **Unit** |
| Input range | 0.0 | 33 | V |
| Input Low | 0.0 | 2.7 | V |
| Input High | 4.1 | 33 | V |

Table : Ignition sense Pin Electrical specifications.

* The input signal will be read as Low (zero) when the level is lower than 2.7VDC.
* The input signal will be read as High (one) when the level is higher than 4.1VDC.

For this configuration we will to use the "Advanced Editor" of LuvitRED.

**Notes**:

1. This document assumes that the reader is familiar with the LuvitRED and the terms explained on the document called Basics\_of\_LuvitRED\_vXXX.pdf.
2. We are using CloudGate firmware version and LuvitRED version for this configuration.

# Setting up the GPIO for Ignition detection.

For this application we are going to use the GPIO in node from LuvitRED.

The "gpio" in node (See Figure 2) is the one that will be in charge of providing us with the Ignition Sense PIN status.



Figure 2: GPIO in node.

Let's start with the configuration:

1. Drag and Drop one "gpio" in node into the LuvitRED editor and double click on it (See Figure 3).

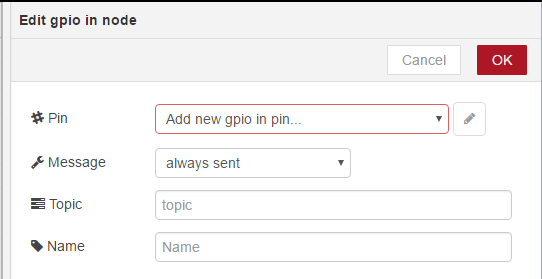


Figure : New gpio in node configuration.

1. The first item that needs to be changed is the “#Pin” configuration. For this, click on the pencil icon to “Add new gpio in pin”:

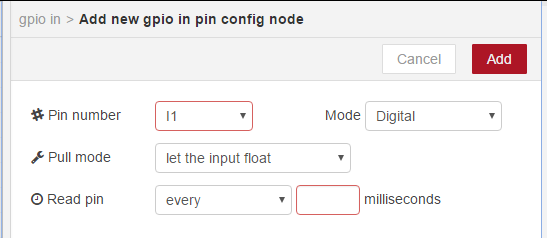


Figure : New gpio in pin default configuration.

We are going to change the standard configuration of the pin as follows:

* #Pin number: IGN
* Mode: Imx

Notice that as soon as the Mode is changed from Digital to Imx, the other options are automatically removed from the configuration page:

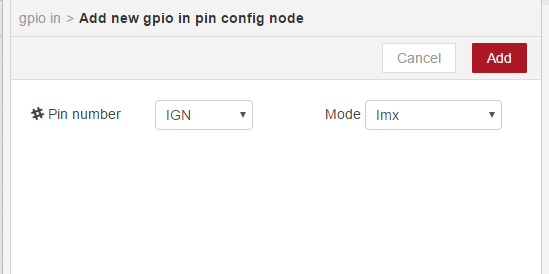


Figure : New gpio in pin configuration for ignition sense.

Click on the “Add” button.

1. After adding the new gpio in pin, change the “Message” from “always sent” to “only sent on change” so that the node only sends a value when the pin changes its state. Change the “Topic” and “Name” of the node to “Ignition” as shown on Figure 6:

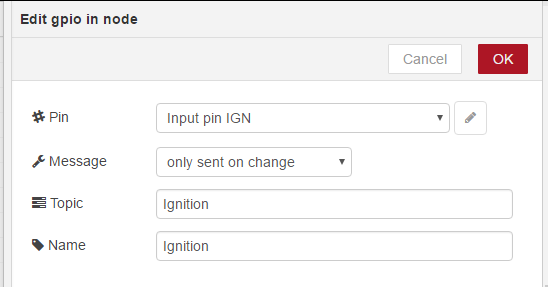


Figure : gpio in node configured.

1. Click on "OK" to close the node configuration.
2. Drag and Drop a “debug” node into the LuvitRED editor and connect the two nodes as shown on Figure 7:



Figure : Nodes connected.

1. Click on “Deploy”.

The configured flow will display the status of the ignition pin as specified on Table 1. On Figure 8 we are showing the ingnition sense pin value chaging from 0 to 1 as we change the ignition possition.

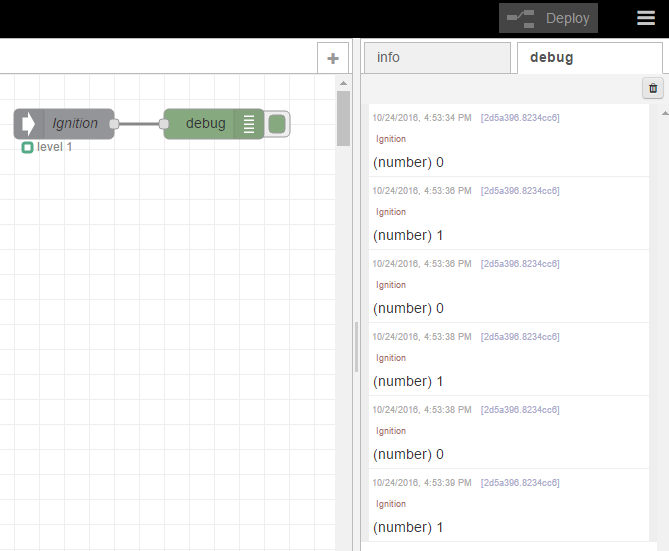


Figure : Ignition sense pin state change.

# Wake up the CloudGate on ignition sense.

The wake up on ignition feature of the CloudGate works by powering down the CloudGate and specifying that it should wake up on ignition detection (ignition sense pin). This type of power down can be achieved using a CloudGate SDK API call named “cg\_status\_t cg\_system\_power\_down”. LuvitRED implements this API call inside the “cloudgate control” node (See Figure 9).



Figure : cloudgate control node.

Let’s start with the configuration:

1. Drag and Drop one "cloudgate control" node into the LuvitRED editor and double click on it (See Figure 10).

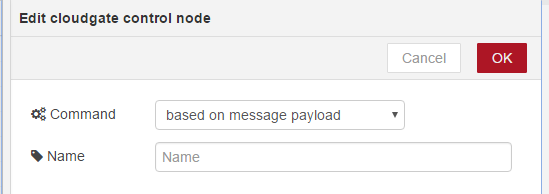


Figure : New cloudgate control node configuration.

1. This node is by default configured to act depending on the incoming message, but it can also be hardcoded to perform an specific action regardless of the incoming message. For simplicity we are going to hardcode it to “power down CloudGate” and to “Arm ignition sense”. Let’s also change the name of the node to “PowerDown\_IGN”:

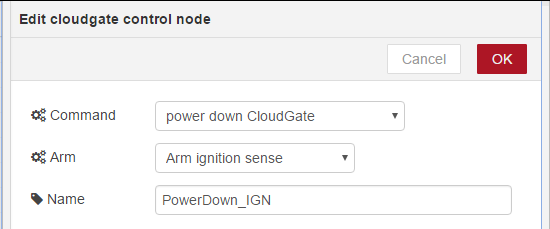


Figure : cloudgate control node configured for wake up on ignition sense.

1. Click on "OK" to close the node configuration.
2. Drag and Drop an “inject” node into the LuvitRED editor and connect the two nodes as shown on Figure 12:



Figure : Nodes connected.

1. Click on “Deploy”.

The configured “PowerDown\_IGN” node will display its status. This status will change from “idle” to “powering down” when we trigger the “inject” node. On Figure 13 and Figure 14 we are showing the two different status of the node. After triggering the “inject” node, the CloudGate will initiate a power down and it will restart when triggering the ignition sense pin (make sure the pin is off (zero) before powering down the CloudGate).



Figure : PowerDown\_IGN before triggering the inject node.



Figure : PowerDown\_IGN after triggering the inject node.

Now that the CloudGate is powered down, go ahead and trigger the ignition to restart the CloudGate.

**NOTE:**

Every time a wake up on ignition sense is desired, the CloudGate needs to be powered down using the right call arming the ignition sense detection, otherwise the ignition will no work for that power down.

# Timed Wake up

The timed wake up feature of the CloudGate works by powering down the CloudGate and specifying that it should wake up after a certain time interval. This type of power down can be achieved using a CloudGate SDK API call named “cg\_status\_t\_ cg system\_set\_timed\_wakeup” in combination with a power down call (See section 3). LuvitRED implements this API call inside the “cloudgate control” node (See Figure 15).



Figure : cloudgate control node.

Let’s start with the configuration:

1. Drag and Drop one "cloudgate control" node into the LuvitRED editor and double click on it (See Figure 16).

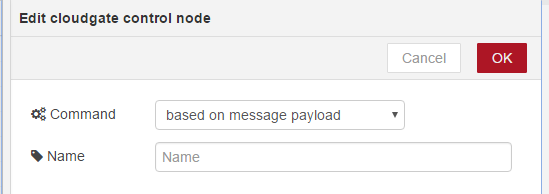


Figure : New cloudgate control node configuration.

1. This node is by default configured to act depending on the incoming message, but it can also be hardcoded to perform an specific action regardless of the incoming message. For simplicity we are going to hardcode it to “power down CloudGate”, “Perform Timed Wait” with a wait of 120 seconds. Let’s also change the name of the node to “PowerDown\_Timed”:

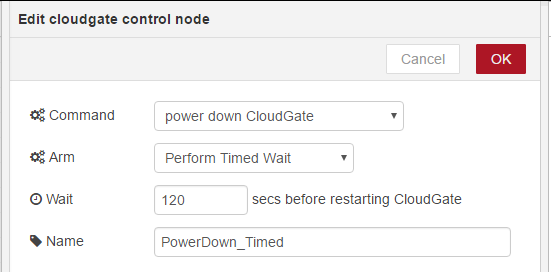


Figure : cloudgate control node configured for timed wake up.

1. Click on "OK" to close the node configuration.
2. Drag and Drop an “inject” node into the LuvitRED editor and connect the two nodes as shown on Figure 18:



Figure : Nodes connected.

1. Click on “Deploy”.

The configured “PowerDown\_Timed” node will display its status. This status will change from “idle” to “powering down” when we trigger the “inject” node. On Figure 19 and Figure 20 we are showing the two different status of the node. After triggering the “inject” node, the CloudGate will initiate a power down and it will restart after the configured time (120s).



Figure : PowerDown\_Timed before triggering the inject node.



Figure : PowerDown\_Timed after triggering the inject node.

**NOTE:**

Every time a timed wake up is desired, the CloudGate needs to be powered down using the right call arming the timed wait, otherwise the internal timer will not work that power down.

# Configuration example

Let’s imagine that we want to have out CloudGate LTE to turn on and turn off based on ignition, but we want it to wait for 1 minute before it goes to the off state.

For this example we are going to reuse the configurations explained on sections 2 and 3. Section 2 will allow us to detect the state of the ignition sense pin and section 3 will allow us to perform the power down and to arm the wake up on ignition feature.

1. Drag, drop and configure the “gpio” in node and the “cloudgate control” nodes as specified on sections 2 and 3 respectively into the LuvitRED editor as shown on Figure 21.

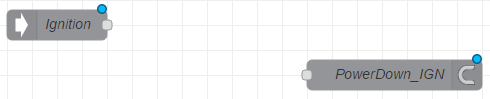


Figure : gpio in and cloudgate control nodes configured.

1. Drag and drop a “switch” node, this node will help us verify if the ignition is tuning off (zero) or not.



Figure : switch node.

* 1. Double click on the node and change the first rule to check it the payload of the message is 0 (zero). The rest can be set to the defaults:

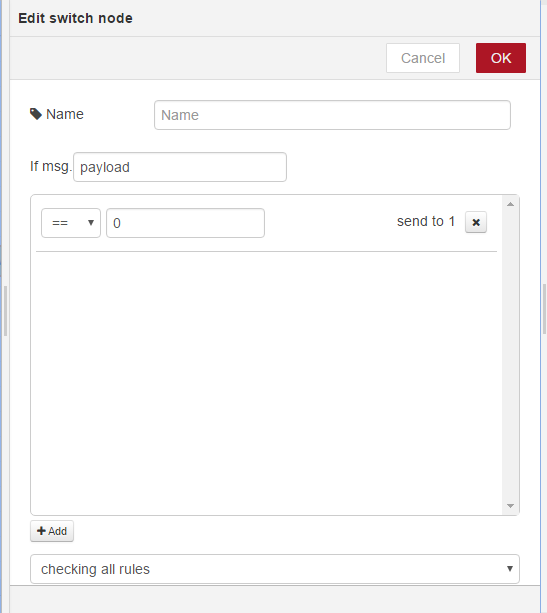


Figure : switch node configuration.

* 1. Click on “OK” to close the node configuration.

1. Drag and drop a “trigger” node, this node will help us add the delay of 1 minute before we power down the device.



Figure : trigger node.

* 1. Double click on the node and change the following:
     + Output: nothing (no output)
     + then wait: 1 Minutes
     + Name: Delay1min

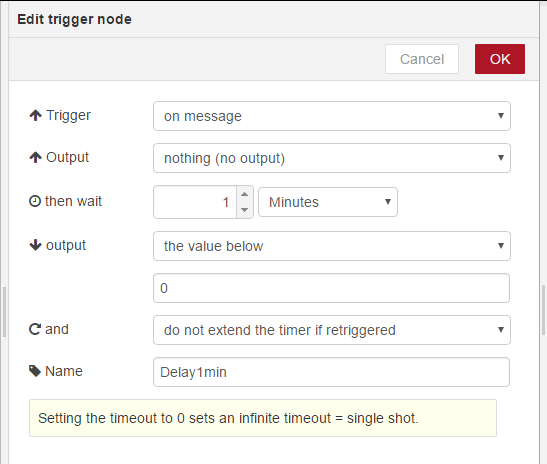


Figure : trigger node configuration.

* 1. Click on “OK” to close the node configuration.

This configuration means that on message the node will not send anything, instead it will wait for a minute before sending a message with a payload equal to 0 (zero).

1. Connect all nodes as shown on Figure 26.



Figure : Nodes connected.

1. Make sure your ignition is on so that the “Ignition” node will detect a 1 on the first moment and click on “Deploy”.

The configuration should look like this at this moment:



Figure : Configuration deployed and ignition sense pin high.

Let’s try our configuration:

1. Turn off the ignition:



Figure : Ignition turned off.

1. After a minute, the “PowerDown\_IGN” node should change from “idle” to “powering down”:



Figure : CloudGate powering down after a minute of the ignition off detection.

1. After ClodGate is down, turn on the ignition. You should see the CloudGate booting up again and go back to the original state shown on Figure 27 waiting for a new ignition off to happen.

**NOTE:**

The switch and trigger nodes, could be replaced by a more complex configuration that takes care of other variables before powering down the CloudGate, for example:

* Wait until all data is sent out.
* Cancel the power down if the driver is turning on the vehicle again.
* Send a report of the actions taken to an email client.
* Others.