Configuring CloudGate mini using LuvitRED

LuvitRED

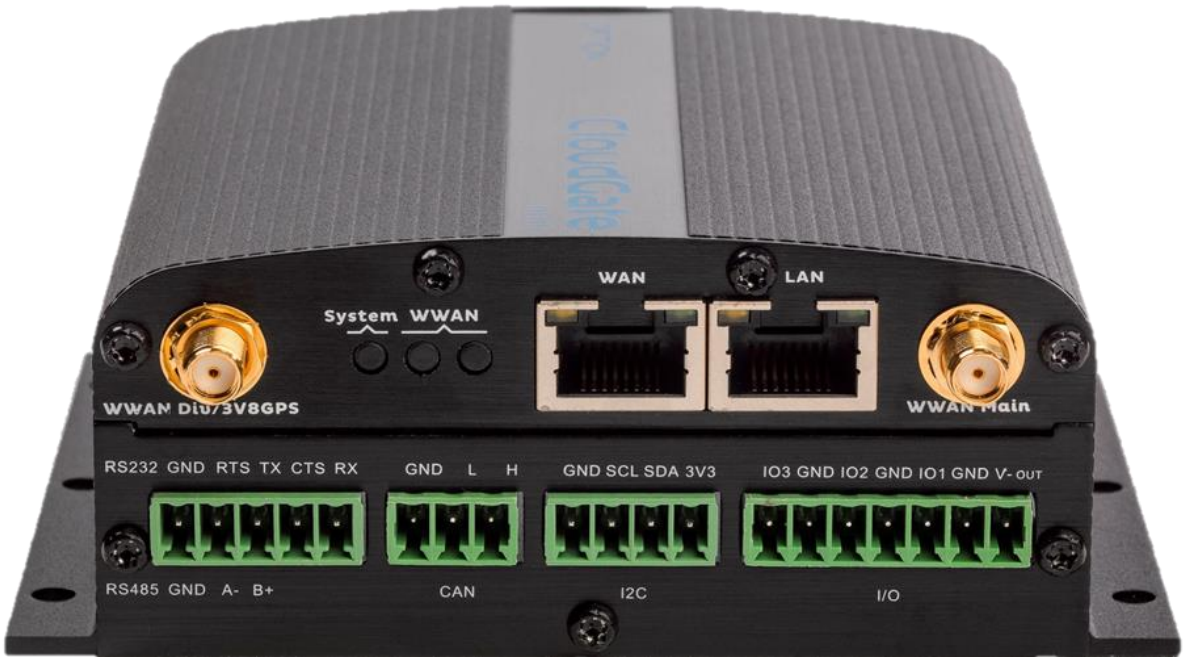
By Harald De Vriendt

OPTION

Table of Contents

Introduction

This document covers the configuration of CloudGate mini's interfaces using LuvitRED. CloudGate mini is a fixed functionality IoT gateway that can be expanded with a WiFi or LoRaWan expansion card in the back slot. The frontside provides a powerful combination of some common IO interfaces that can be used to connect your physical sensors, devices or tools to the cloud.



From left to right CloudGate mini has a:

- Serial interface (RS232/RS485)
- CAN bus
- I2C
- GPIO+V-out

LuvitRED

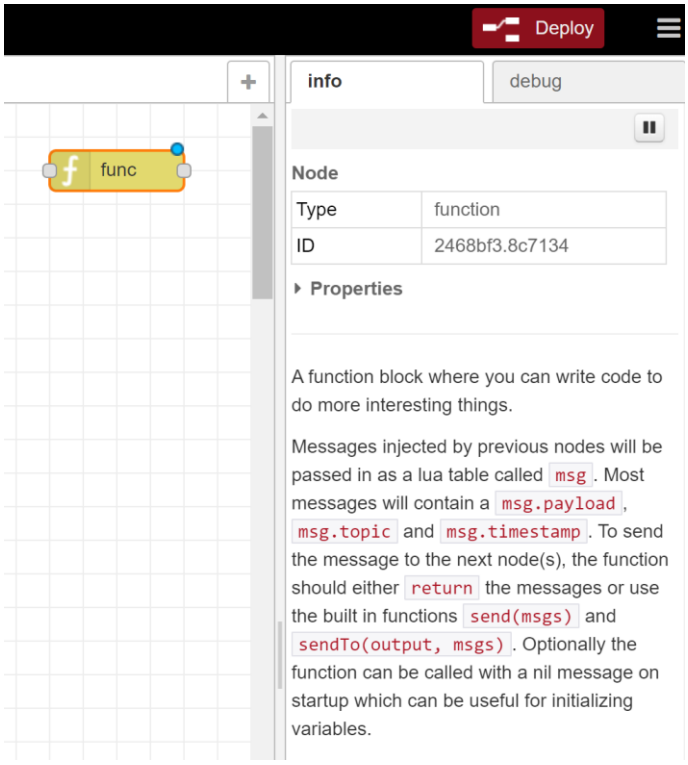
The glue between these physical interfaces and the cloud or even local processing and decision making is LuvitRED.

The LuvitRED editor is accessible through the CloudGate webUI. Following instructions assumes basic knowledge on how to use and access the LuvitRED editor. If you have not yet installed LuvitRED, please follow these steps first. https://cloudgateuniverse.com/docs/video-tutorials

You can find more information about LuvitRED here:
https://cloudgateuniverse.com/docs/luvitred

Info Tab

The info Tab is a very powerful tool in LuvitRED and is key for development. You can find it on the right side of the screen, next to the debug tab.



When you click on a node, it displays all documentation for that node and the configuration screen you are currently in. This includes the attributes you can set, messages it accepts and the output from this node.

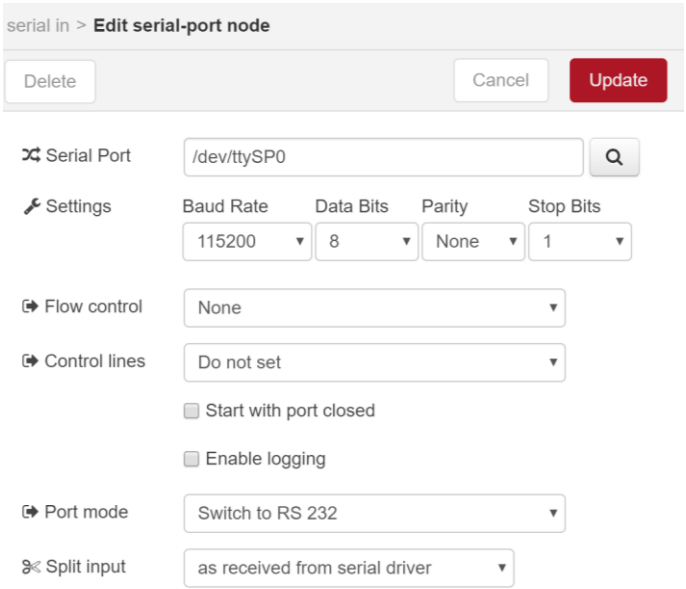
Serial Interface

The serial interface can be used in different modes. These modes are set in the serial config node.

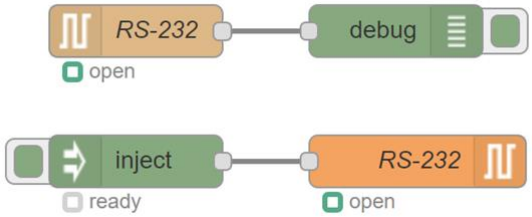
To configure the serial node, double click the serial in- or output node and select the config in the drop-down menu. If you have not created a config yet, click the pencil and fill out the wanted settings. Both RS232 and RS485 are available on port */dev/ttySP0*.

RS232

To enable RS232, set the port mode to 'Switch to RS 232'. Select the baud rate, data bits, parity and stop bits to your needs. These settings depend on the device you need to communicate to. You can do this for both the in- and out nodes.

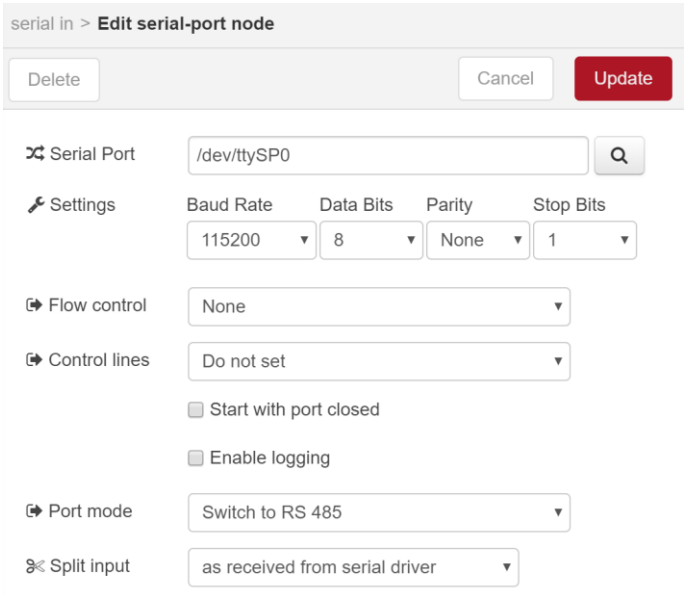


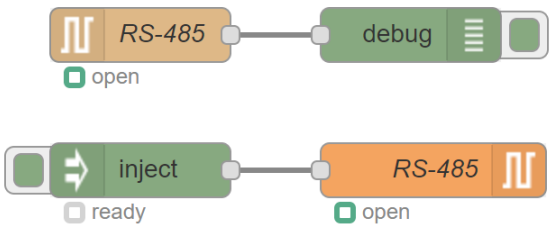
After deploying this, data received over RS-232 will be spit out of the node. For debugging you can connect a debug node to the RS-232 in node, or an inject to the RS-232 out node. This way we can see the received data on the debug screen and inject a timestamp message.



RS485

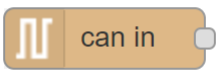
The procedure to enable RS485 is almost identical to using RS232, set the port mode to 'Switch to RS 485' or RS 485 Terminated (depending on your bus). Select the baud rate, data bits, parity and stop bits to your needs. These settings depend on the device you need to communicate to.





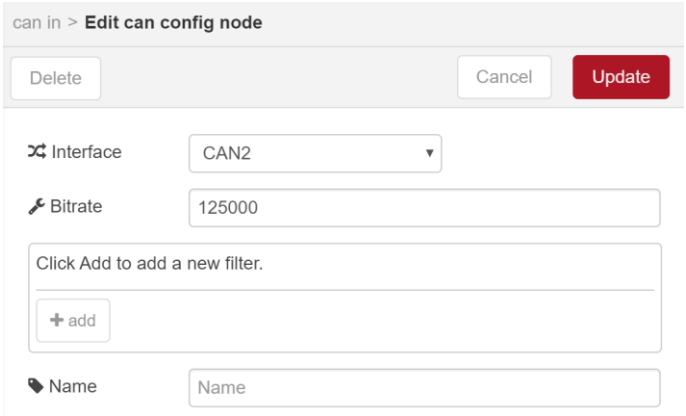
CAN

The CAN interface on CloudGate can be read out using the CAN node.



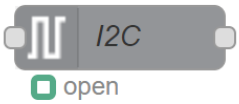
Double click the CAN node and click the pencil to add a can config. Here you can set the bitrate and add filters. Higher speeds than 550 kbps are not guaranteed. A filter matches when <received_can_id> & mask == can_id & mask. More information can be found in the 'info' tab in LuvitRED.

Important to note: If you do not set any filters, no data will come in.

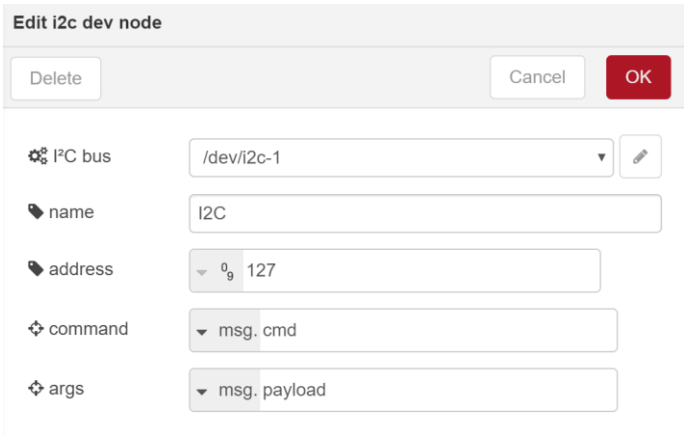


I2C

The I2C node allows you to connect to an I2C sensor. Note that the I2C wires cannot be too long and that I2C is only made to function up to a few centimeters.



You can configure the I2C device, */dev/i2c-1*, the address of your sensor and from what messages the node can be controlled. You can only connect one sensor on this connector. The address and registers of your sensor can be found in the sensor's datasheet.



After configuring the node, it can be controlled by inserting a command. Inserting a message with:

msg.cmd = 'read'
msg.payload = 100

Will read 100 bytes from the device address specified in the node.

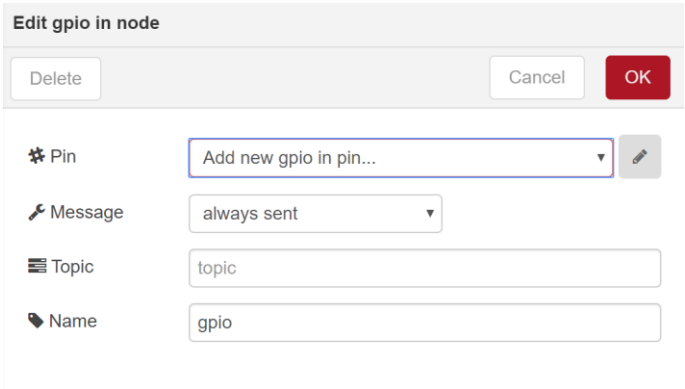
To insert these commands, an inject node together with a function block can be used. In the function block, the content of the msg object will be modified to contain the command and payload. For demonstration, the output can be sent to a debug node.

IO

There are three IO pins available on the IO connector. Each one is accompanied by a GND pin. There is also one V-Out and GND pair available.

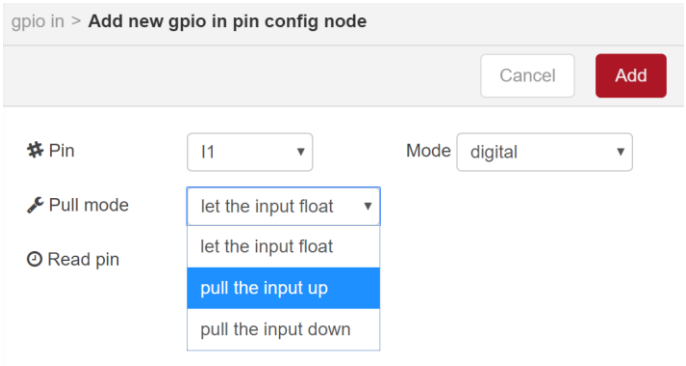
Each IO pin can be configured as:

- Digital In (float / pull up / pull down)
- Digital Out (push pull / open drain
- Pulse count
- Analog In
- Current loop



GPIO In

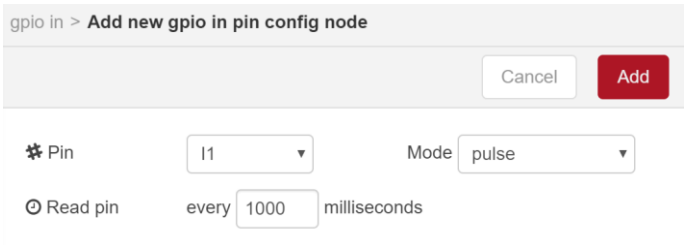
Digital Input



(0 - 30V)

The digital input emits the state of the pin at a specific interval or on change. This value is emitted in the msg.payload field as a boolean (True/False).

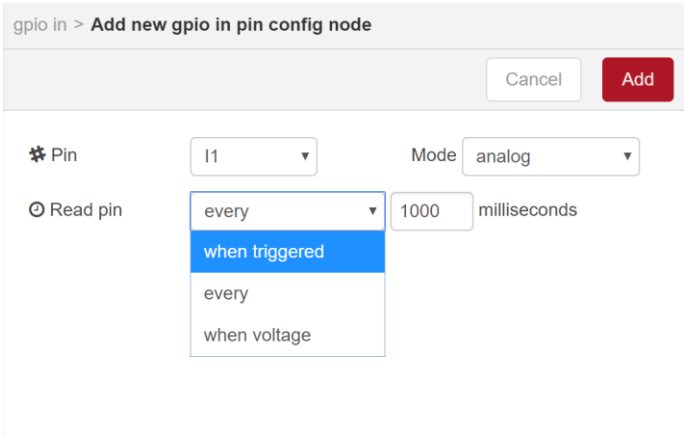
Pulse Count (S0)



(0 - 30V)
(Max frequency: 1000Hz)

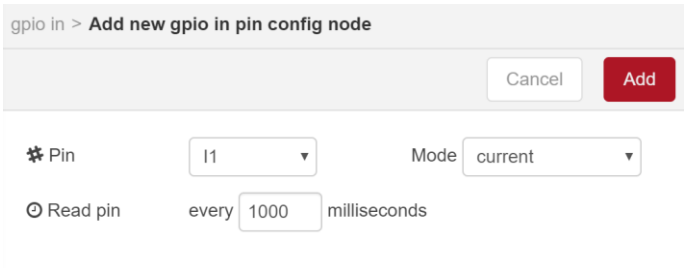
The pulse counter (S0) is a mode of the GPIO In node that counts the pulses presented on the input pin. The stored value is an uint32. Pulse counters are primarily used to measure (electricity) consumption. Please note that some pulses might not be counted during a reboot and initialization phase of the device. You can use the number of pulses to calculate a consumption over time. Therefor it is key to know the unit that your pulse resembles.

Analog Input



(0 - 3.3V)

CL Input



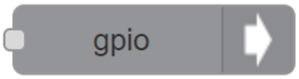
(4 – 20mA)

The value received in LuvitRED is the measurement of the voltage over a resistor inside the CloudGate. The output value in LuvitRED will be the voltage level measured over a resistor inside the CloudGate. To get a value between 0 and 20 which correspond to the measured sensor value. If a zero (0) is measured, this means a wire break in the sensor connection. You can map the measured value to a percentage using following formula:

$$percentage = \frac{(value - 4)}{16} * 100$$

GPIO Out

GPIO out node can be set as a digital output.



Inverting the output in the GPIO out settings reverses the functioning of the output.

Digital Output

Push-pull (0 – 3.3V)
Open drain (< 250mA, 0 – 30V)

The digital output node listens for messages containing a boolean payload to change the output state of the configured pin. The current state is displayed underneath the node. The pin can be configured as push/pull or as open drain in the GPIO config node.