Node-RED & Blynk IoT Integration Guide

This document provides a complete, step-by-step guide to integrate Node-RED using the node-red-contrib-blynk-iot nodes with Blynk Cloud. It matches the Node-RED flow shared earlier, so you can control relays and monitor inputs using Blynk assets like switches and LEDs. There are other examples as well shared in the node red examples folder. For demonstration purposes this guide specifically caters to IoTextra Octal2 board, but can be directly applied to other Analog, Digital and Combo IoTextra boards as general principles are the same.

---

1) Create a Template and Device in Blynk.Cloud

1. Log in to **Blynk.Cloud** (Console / Developer zone).
2. Create a **Template** (give it a name, select hardware type such as "Other" and connection type `wifi`).
3. Add the Datastreams (virtual pins you'll use: V0, V1, V2 …) and widgets to the Web dashboard / App if desired.
4. **Create a Device** from that template. After creating the device, you will see the **Template ID** and **Device Auth Token** (take note / copy them). You'll need both to configure Node-RED. Blynk Documentation+1

Tip: Developer Mode must be on in your Console to access template/dev tools.

Example Blynk Cloud Setup

1. Log in to https://blynk.cloud
2. Developer Zone → Templates → + New Template
   - Hardware: Other
   - Connection: WiFi
3. Add 8 Virtual Datastreams:

Name	Virtual Pin	Data Type	Widget
Relay 1	V0	Integer 0/1	Switch
Relay 2	V1	Integer 0/1	Switch
Relay 3	V2	Integer 0/1	Switch
Relay 4	V3	Integer 0/1	Switch
Input 1	V4	Integer 0/1	LED
Input 2	V5	Integer 0/1	LED
Input 3	V6	Integer 0/1	LED
Input 4	V7	Integer 0/1	LED

4. In the Web Dashboard or mobile app, add **Switch** widgets for V0–V3 and **LED** widgets for V4–V7.
5. Save the template.
6. Create a Device from the template and copy the **Template ID** and **Device Auth Token**.

---

2) Install `node-red-contrib-blynk-iot` into Node-RED

From the machine running Node-RED:

Option A — Palette Manager (UI):

- Open the Node-RED editor → top right menu → Manage palette → Install → search `node-red-contrib-blynk-iot` → Install.

Option B — command line (if Node-RED installed globally):

```
# inside your Node-RED user folder (~/.node-red) or globally if you run
Node-RED globally:
npm install node-red-contrib-blynk-iot
# or globally:
npm install --global node-red-contrib-blynk-iot
```

After install, **restart Node-RED** so the new nodes appear.

---

3) Add & configure the Blynk client (config node) in Node-RED

- In the Node-RED palette you'll now see **blynk-iot-client** (a *config* node) plus blynk widgets (blynk-iot-in-write, blynk-iot-out-write)
- Double-click the **blynk-iot-client** nodes you want to use and add it to your flow.
- Server: `blynk.cloud` (or your regional host like `fra1.blynk.cloud`)
- Template ID: paste from Blynk
- Auth Token: paste from Blynk
- **Name**: whatever you like (e.g. `BlynkCloud-MyDevice`).
- Deploy the flow.

---

4) Build a **minimal** test flow (send → cloud, cloud → Node-RED)

Goal: demonstrate both directions.

What to drag (minimal):

- `Our mezzanine` nodes from before (test input/outputs)
- `blynk-iot-out-write` node (to write a virtual pin in Blynk)
- `blynk-iot-in-write` node (to receive writes from the Blynk cloud)
- `change` node (to change the message payload into a logic the node-red flow requires)
- `debug` node (optional)

If in doubt of above steps look at examples files.

Note: each blynk node has a **help** panel (double-click node → Help), check it for the exact message structure (msg.payload, msg.pin etc.). The contrib author added a help panel on each node.

Example Test Flow:

- Toggle a switch in Blynk: the Node-RED dashboard switch changes and the MQTT channel triggers.
- Change an MQTT input: the Node-RED LED updates and the Blynk LED reflects the change.

---

5) Message formats, quick rules

- **Outgoing (Node-RED → Blynk)**
  - If you configured the blynk-out node with a Pin (e.g. `0`), set `msg.payload` to the value you want to write (number/string/Boolean as per datastream type).
  - Some nodes also accept `msg.pin` or `topic` overrides, check node help panel for the exact supported fields. Node-RED Library
- **Incoming (Blynk → Node-RED)**
  - `blynk-iot-in-write` typically outputs `msg.payload` containing the new value and often `msg.pin` or similar to identify which virtual pin triggered it, again check the node help panel for the exact structure.

---

6) Tips

- Ensure Node-RED can reach the internet and the blynk.cloud.
- If nothing appears in Node-RED debug when you change the Blynk widget: confirm the **Device Auth Token** is correct and that the node's **client** config uses that token. Also confirm the Virtual Pin numbers match.
- Use the regional server shown in Blynk Console if provided.
- Datastream types must be the same between node-red and blynk (Example: **Integer (0/1)**) to match the Change nodes in the flow.
- If you prefer MQTT or want to integrate other systems, Blynk also provides an MQTT broker option (server blynk.cloud:8883, username device, password = device Auth Token), Blynk docs include a full Node-RED MQTT example (subscribe to downlink/ds/# and publish to ds/<DatastreamName>). Blynk Documentation+1

Useful links (official)

- node-red-contrib-blynk-iot Node-RED package page (install + node list). Node-RED Library
- Blynk docs — Node-RED / MQTT integration guide (contains example Node-RED flows and MQTT topic formats). Blynk Documentation
- Blynk docs — Device activation / getting Auth Token & Template ID. Blynk Documentation+1