

External Notification and Canned Messages

External Notification Module

The External Notification module triggers a visual or audio alert on the node hardware when a message is received - useful for heads-up awareness without constantly watching a screen.

Configuration

```
meshtastic --set external_notification.enabled true
meshtastic --set external_notification.output_ms 1000
meshtastic --set external_notification.active true
meshtastic --set external_notification.alert_message true
```

`output_ms` is how long to activate the output in milliseconds. `active` sets the logic level (true = active high output). `alert_message` triggers on incoming text messages; `alert_bell` triggers on all message types.

Hardware Outputs

The module drives a GPIO pin that can trigger:

- **Buzzer** - Active buzzer connected between the GPIO pin and GND (through a transistor for modules requiring more current)
- **LED** - External LED indicator, useful for daytime visibility or status display
- **Relay** - Via a relay module, can trigger any external device: alarm siren, strobe light, or notification device

Most T-Beam and Heltec boards have accessible GPIO pins; consult your board's pinout documentation for safe GPIO numbers (avoid pins used by LoRa, I2C, and UART).

Canned Messages Module

The Canned Messages module allows quick message sending from the device hardware without using the phone app - ideal for when you have a dedicated node with a small rotary encoder or button input.

Use Case

Deploy a node in a fixed location (workshop, vehicle, equipment room) with a rotary encoder. The user can navigate a pre-programmed list of messages and send them with a button press, without needing to open a phone app. Useful for sending routine status messages in situations where opening a phone is inconvenient.

Configuration

```
meshtastic --set canned_message.enabled true
meshtastic --set canned_message.messages "OK|On my way|Need help|ETA 10 min|Stand by"
```

Messages are separated by pipe characters. Maximum total length is limited by protobuf message size; keep individual messages short.

Input Hardware

Requires a rotary encoder connected to GPIO pins, or can use UP/DOWN/SELECT button inputs. Configure the GPIO pin assignments:

```
meshtastic --set canned_message.inputbroker_pin_a 39
meshtastic --set canned_message.inputbroker_pin_b 40
meshtastic --set canned_message.inputbroker_event_cw MSG_INPUT_EVENT_NEXT
meshtastic --set canned_message.inputbroker_event_ccw MSG_INPUT_EVENT_PREV
```

Pin numbers are board-specific. The RAK WisBlock Input module provides a pre-built button/encoder input board compatible with the WisBlock system.

Revision #2

Created 2026-05-03 05:41:27 UTC by Mesh America Admin

Updated 2026-05-03 12:59:18 UTC by Mesh America Admin