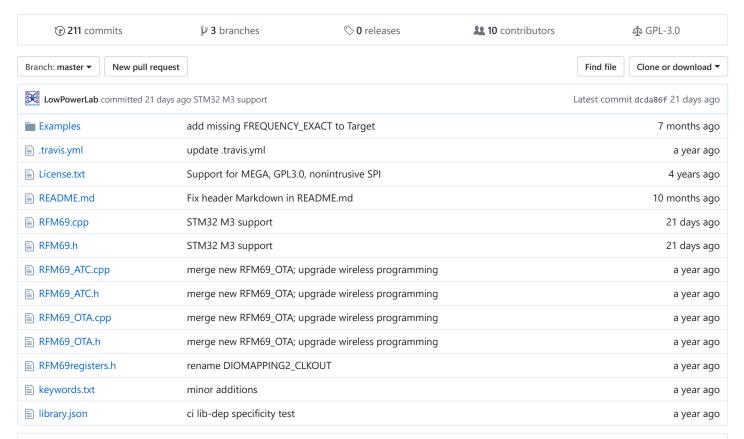🖵 **LowPowerLab** / **RFM69**

Join GitHub today

Dismiss

GitHub is home to over 20 million developers working together to host
and review code, manage projects, and build software together.

Sign up

RFM69 library for RFM69W, RFM69HW, RFM69CW, RFM69HCW (semtech SX1231, SX1231H)

ⓣ **211** commits	⑂ **3** branches	🏷 **0** releases	👥 **10** contributors	⚖ GPL-3.0

Branch: master ▾ | New pull request | | Find file | Clone or download ▾

🖽 **LowPowerLab** committed 21 days ago STM32 M3 support   Latest commit `dcda86f` 21 days ago

📁 Examples	add missing FREQUENCY_EXACT to Target	7 months ago
📄 .travis.yml	update .travis.yml	a year ago
📄 License.txt	Support for MEGA, GPL3.0, nonintrusive SPI	4 years ago
📄 README.md	Fix header Markdown in README.md	10 months ago
📄 RFM69.cpp	STM32 M3 support	21 days ago
📄 RFM69.h	STM32 M3 support	21 days ago
📄 RFM69_ATC.cpp	merge new RFM69_OTA; upgrade wireless programming	a year ago
📄 RFM69_ATC.h	merge new RFM69_OTA; upgrade wireless programming	a year ago
📄 RFM69_OTA.cpp	merge new RFM69_OTA; upgrade wireless programming	a year ago
📄 RFM69_OTA.h	merge new RFM69_OTA; upgrade wireless programming	a year ago
📄 RFM69registers.h	rename DIOMAPPING2_CLKOUT	a year ago
📄 keywords.txt	minor additions	a year ago
📄 library.json	ci lib-dep specificity test	a year ago

📖 **README.md**

RFM69 Library `build passing`

By Felix Rusu, LowPowerLab.com
RFM69 library for RFM69W, RFM69HW, RFM69CW, RFM69HCW (semtech SX1231, SX1231H)
The latest examples, new features and bug fixes are found in the original repository of this library.

License

GPL 3.0, please see the License.txt file for details. Be sure to include the same license with any fork or redistribution of this library.

Features

- easy to use API with a few simple functions for basic usage
- 255 possible nodes on 256 possible networks
- 61 bytes max message length (limited to 61 to support AES hardware encryption)
- customizable transmit power (32 levels) for low-power transmission control
- sleep function for power saving
- automatic ACKs with the sendWithRetry() function
- hardware 128bit AES encryption
- hardware preamble, synch recognition and CRC check
- digital RSSI can be read at any time with readRSSI()
- interrupt driven
- tested on Moteino R3, R4, R4-USB (ATMega328p), MEGA (ATMega1284p)
- works with RFM69W, RFM69HW, RFM69CW, RFM69HCW, Semtech SX1231/SX1231H transceivers
- promiscuous mode allows any node to listen to any packet on same network

Library Installation (Arduino IDE)

Copy the content of this library in the "Arduino/libraries/RFM69" folder.
To find your Arduino folder go to File>Preferences in the Arduino IDE.
See this tutorial on Arduino libraries.

Hardware & programming

The easiest way to get started is with the well documented and supported Moteino microcontroller platform which is easily programmable from the Arduino IDE. This includes the Moteino, MoteinoUSB & MoteinoMEGA. RFM69 transceivers were extensively tested on Moteinos for the purpose of building internet of things (IoT) devices that can be controlled wirelessly. This platform has matured over time and there is now a dedicated page where you can review how these devices can interact with each other via a RaspberryPi gateway interface. Here's a video overview:
https://www.youtube.com/watch?v=YUUZ6k0pBHg
https://www.youtube.com/watch?v=I9MNZQgqKHA
https://www.youtube.com/watch?v=F15dEqZ4pMM

Basic sample usage

- The Gateway example listens for incoming data from remote nodes and responds to any ACK requests
- The Node example is a loop that sends increasingly longer packets to the gateway and waits for an ACK each time
- More examples are added from time to time, check all the examples, visit the LowPowerLab blog for latest news and projects, and check out the LowPowerLab forums for projects and discussion

Blog writeup

See the library release blog post

Why RFM69

- I have spent a lot of time developing this library for RFM69W/HW transceivers. I made it open source because I believe a lot of people can benefit from this new powerful transceiver. I hope others will also contribute and build on my work
- I have long researched alternative transceivers for RFM12B which is still an excellent transceiver but it is much lower output power and has limited built in features which need to be implemented in firmware (PREAMBLE, SYNC, CRC, packet engine, encryption etc).
- I wanted a transceiver that could still be very small, easy to use, and have the longer range that I needed
- RFM69 comes in 2 variants that have the same pinout layout: RFM69W (13dBm, 45mA TX) and RFM69HW (20dBm, 130mA TX). Other variants include the RFM69CW (up to 13dBm power) which is pin compatible with RFM12B, and RFM69HCW (20dBm output power).

RFM69W range and antennas

- I have tested open-air range on these transceivers in various combinations.
- I am happy to say that a range of upwards of 350m can be achieved with the default settings provided in the library. Some users reported upwards of 500m by lowering the bitrate, and a forum user reported 1.5miles at 1.2Kbps: see this forum post and this blog page
- The caveat with these higher RF power units is that they need more DC power when they transmit. For battery powered motes, you will need to keep them powered down and only transmit periodically. Use the sleep() function to put the radios in low power mode and use the LowPower or Narcoleptic libraries to power down your Moteino/Arduino