



Instructions for installation of 125KHz Reader software

Provisional
Jan 2023

The purpose of our software is to demonstrate the basic capabilities of RFID-reading hardware and to provide some working software as a platform for starting project development. Provided are three terminal applications in C and Python respectively, and one GUI application written in Python.

This software is experimental and is not intended to be used in a production environment. The source code is available at: www.github.com/CognIoT

Our software is free; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation as version 2 of the License. For more information refer to: www.cogniot.eu

USAGE & FUNCTIONALITY

- H1/S and H2 Hitags have read/write functionality, with the capacity to do so with both individual tag data pages, and with tag data blocks (multiple data pages on the RFID Tag).
- EM/MC2000 tags do not support Page data writing and Block read/write functionality.
- For optimal usage place the tag to the antennae before sending a command to the program.

COMMANDS FOR TERMINAL PROGRAMS

Using the supplied software the following functions are available, each command being accessed by the letter.

```
z - Display firmware version information
S - Acknowledge presence of Tag
F - Perform a Factory Reset
P - Program EEPROM Polling delay
v - Select reader operating mode
R - Read Tag and PAGE 00 data
r - Read Tag and BLOCK 04 data
W - Write Tag and PAGE of data
w - Write Tag and BLOCK of data
A - Read ALL Pages 0 - 3f
a - Read ALL Blocks 0 - 16
e - Exit program
```

DEPENDENCIES

This software uses the following Dependencies:

- **Tk interface:** a graphical user interface framework installed with all Python distributions. Read the documentation here: <http://docs.python.org/3/library/tkinter.html>
- **WiringPi:** a GPIO access library for use with the Raspberry Pi. Designed for use with C and RTB (BASIC), but adaptable to other languages. As of 2022 it is compatible with the Raspberry Pi 4B. Read the documentation here: <http://wiringpi.com>
- **pySerial:** a serial port access library for Python: <https://pypi.org/project/pyserial/>

INSTALLATION INSTRUCTIONS

To install the software, follow these simple steps:

1. Upgrade and Update the operating system via the terminal:
 - a. `sudo apt-get update`
 - b. `sudo apt-get upgrade`
2. Using Raspberry Pi Configuration, disable the shell and kernel from using the serial port:
 - a. To do this from within the Raspberry Pi OS:
 - i. From the Menu, select Preferences → Raspberry Pi Configuration
 - ii. On the interfaces tab, set Serial to 'Disabled'.
 - b. To do this via the terminal:
 - i. `sudo raspi-config`
 - ii. Interface Options → Serial Port → Disable serial login shell & serial interface.
3. Install the necessary software to support the CognIoT demonstration application software:
 1. Install Python Tools:
 - a. `sudo apt-get install python-dev python-setuptools`
 2. Install pySerial:
 - a. `pip install pyserial`
 3. Install WiringPi:
 - a. `sudo apt-get install git-core`
 - b. Check whether WiringPi is already installed:
 - i. `gpio -v`
 - c. If there is a version present:
 - i. `sudo apt-get purge wiringpi`
 - ii. `hash -r`
 - d. Install a fresh version of WiringPi:
 - i. `git clone https://github.com/WiringPi/WiringPi.git`
 - ii. `cd WiringPi`

- iii. `git pull origin`
- e. Build WiringPi and check version for success:
 - i. `./Build`
 - ii. `gpio -v`
 - iii. `gpio readall`

4. Install this CognIoT software:

- a. `git clone https://github.com/CognIoT/RFID_125kHz`

5. Two C and Python terminal programs, as well as a GUI application built in Python, can now be found in the newly created RFID_125kHz folder.

- a. `cd RFID_125kHz`

6. To run Python terminal script, switch to the python directory and run it:

- a. `cd python`
- b. `sudo python3 RFIDReader.py`

7. To run the C terminal program, it first needs to be compiled:

- a. `cd c`
- b. `./build`
- c. It can be run with:
- d. `sudo ./RFIDReader`

8. You can also now run the Python-based GUI or USB scripts from within your Raspberry Pi:

- a. Locate the folder from within your Raspberry Pi's file explorer.
- b. Right click and execute the Python GUI file.