



## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

---

// oude afbeelding





## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

### Document Information

<b>Title</b>	ERF3000v2
<b>Subtitle</b>	LTE Cat. M1, Cat. NB1, EGPRS & GNSS Arduino Shield
<b>Document type</b>	User Guide
<b>Document number</b>	1
<b>Revision and date</b>	R1      12-06-2019

### Product status

In production

this document applies to the following products:

Name	Type number	version	Product status
ERF3000	ERF3000	1.5	EOL
ERF3000v2	ERF3000	1.0	In production



## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

---

### Contents

Introduction .....	4
Set-up .....	4
1 Getting a Network connection .....	6
1.1 AT – commands .....	6
1.2 Examples .....	7
2 Firmware update .....	8
2.1 USB .....	8
2.2 DFOTA .....	10
3 Creating a Low-Power application .....	11
3.1 PSM .....	11
3.2 eDRX .....	11
3.3 Antenna design .....	11
3.3.1 Antenna design tuning .....	14
4 Arduino Programming .....	15
4.1 Setup .....	15
4.2 Example .....	15
5 Measuring the current consumption .....	16
5.1 Preparations .....	16
5.2 Results .....	16



## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

### Introduction

This document functions as a Quick start guide / User guide. Several basic processes and features are described in the document to help speed up the development and evaluation time. Several of these basic processes are: Getting a network connection, performing a firmware update, creating an Arduino software example and explaining how to create a Low power Internet Of Things (IoT) application.

### Set-up

To start evaluating the shield, please follow below steps:

Step 1: Connect the GSM antenna.

Step 2: Plug in a Nano SIM card.

Step 3: Plug in a USB mini cable and connect it to a pc/laptop.

Step 4: Plug in the power adapter supplied with the shield.

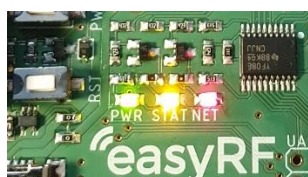
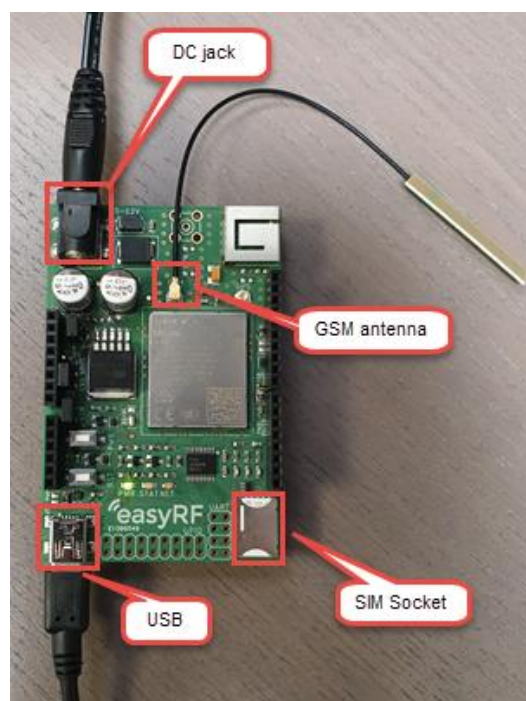


Please see chapter 3.1 Power of the ERF3000 Datasheet when using a different adapter.

Step 5: When power is supplied to the shield the green LED marked with PWR should light up.

Step 6: Press the PWR button to start the shield.

Step 7: After the module is done with starting up, the yellow LED marked with STAT should light up, and the red LED marked with NET will start blinking.



When above steps are followed, the shield is ready to receive AT-commands. Before a serial connection to the shield can be opened, the correct drivers need to be installed on your device.

Please download them from the following link: <https://www.quectel.com/ProductDownload/BG96.html>

When the download is done, open and extract the .zip file.






## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

Inside the BG96 folder navigate to the “Driver” folder, and open the Quectel\_LTE\_Windows\_USB\_Driver\_V2.0.zip. inside this .zip folder you can find a setup.exe file, install this on you device.

When the installation is done, the PC will recognize the shield.

To verify this go to, device management and look for the following 3 ports.

-  Quectel USB AT Port (COM5)
-  Quectel USB DM Port (COM3)
-  Quectel USB NMEA Port (COM4)

Now any serial software like Putty, Realterm, etc.. can be used to send AT-commands to the BG96.

Quectel also developed special tools for their modules, these are called Qnavigator and QCOM. These programs can also be found in the BG96 folder in the “Tool” folder.

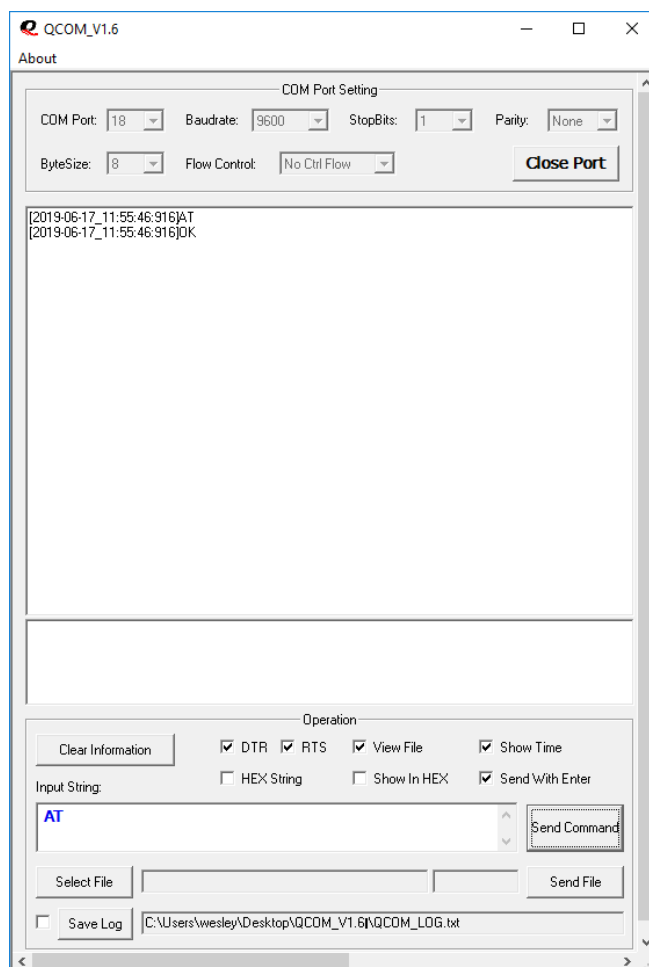
- QCOM has the option to write scripts that will be send to the BG96 module.
- QNavigator is a dashboard environment with some examples.

Open either of the programs and set the COM Port settings to:

- 115200 baud
- Stopbits: 1
- Parity: none
- Bytesize: 8
- Flow control: no flow control

Set the COM Port to the USB AT Port, the number of the port can be found in your device management.

Now open the port and send AT, the module should respond with OK.





## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

### 1 Getting a Network connection

#### 1.1 AT – commands

In order to establish a connection, the module must be initialized with AT-commands. Before attempting to connect to a network, the module must be initialized. Please note, most providers use different settings for all below commands. See chapter [1.2 Examples](#) for examples of the Dutch network providers.

Command	Description
AT+CFUN=0	Will disable all phone functionality of the module

When the phone functionality is disabled please set below settings.

Command	Description
AT+QCFG="band"	This command specifies the frequency bands allowed to be searched of UE
AT+QCFG="nwscanseq"	This command specifies the searching sequence of RATs
AT+QCFG="nwscanmode"	This command specifies the RAT(s) allowed to be searched
AT+QCFG="iotopmode"	This command specifies the network category to be searched under LTE RAT
AT+QCFG="nbsibscramble"	This command specifies the scramble function
AT+CGDCONT	This command specifies the PDP context parameters for a specific context

When the commands are set, the phone functionality can be activated again, and a connection to the network can be forced.



Before forcing a connection to a network, please check with your local provider if IMEI registration of the Quectel hardware is necessary. If you don't register the IMEI number with you provider and still connect to their network you may be put on a blacklist, and access to the network will be blocked. The IMEI number can be found on the metal casing of the Quectel module.

Command	Description
AT+CFUN=1,1	Will activate all phone functionality of the module and reset the module
AT+COPS	This command forces an attempt to select and register the network operator

When AT+COPS returns OK. A connection has been established. Information about the connection can be gained with the following commands:

Command	Description
AT+QNWINFO	This command indicates network information such as the access technology selected, the operator, and the selected band.
AT+QSPN	This command will display the name of the registered network
AT+CSQ	This command indicates the received signal strength (RSSI) and the channel bit error rate
AT+QCSQ	This command is used to query and report the signal strength of the current network service. Dependent on the network this will return: RSSI, RSRP, SINR & RSRQ
AT+QENG="neighbourcell"	This command will return the neighboring cell towers information
AT+QENG="servingcell"	This command will return the serving cell tower information



## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

### 1.2 Examples

Below are examples of network settings for all 3 major Dutch network providers:

<b>KPN NL</b>
<b>CAT-M Commands</b>
AT+QCFG="band",1,80000,1,1
AT+QCFG="nwscanseq",020202,1
AT+QCFG="nwscanmode",3,1
AT+QCFG="iotopmode",2,1
AT+QCFG="nbsibscramble",0
AT+CGDCONT=1,"IP","item.webtrial.m2m"
AT+COPS=1,2,"20408",8

<b>T-Mobile NL</b>	
<b>CAT-M Commands</b>	<b>NB-IoT Commands</b>
AT+QCFG="band",1,4,80,1	AT+QCFG="band",1,4,80,1
AT+QCFG="nwscanseq",030303,1	AT+QCFG="nwscanseq",030303,1
AT+QCFG="nwscanmode",3,1	AT+QCFG="nwscanmode",3,1
AT+QCFG="iotopmode",2,1	AT+QCFG="iotopmode",2,1
AT+QCFG="nbsibscramble",0	AT+QCFG="nbsibscramble",0
AT+CGDCONT=1,"IP","SMARTSITES.T-MOBILE"	AT+CGDCONT=1,"IP","cdp.iot.t-mobile.nl"
AT+COPS=1,2,"20416",8	AT+COPS=1,2,"20416"

<b>VodafoneZiggo NL</b>	
<b>CAT-M Commands</b>	<b>NB-IoT Commands</b>
AT+QCFG="band",1,80000,80000,1	AT+QCFG="band",1,80000,80000,1
AT+QCFG="nwscanseq",020202,1	AT+QCFG="nwscanseq",030303,1
AT+QCFG="nwscanmode",3,1	AT+QCFG="nwscanmode",3,1
AT+QCFG="iotopmode",2,1	AT+QCFG="iotopmode",2,1
AT+QCFG="nbsibscramble",0	AT+QCFG="nbsibscramble",0
AT+CGDCONT=1,"IP","live.vodafone.com"	AT+CGDCONT=1,"IP","nb.inetd.gdsp"
AT+COPS=1,2,"20404",8	AT+COPS=1,2,"20404",9



Before forcing a connection to a network, please check with your local provider if IMEI registration of the Quectel hardware is necessary. If you don't register the IMEI number with you provider and still connect to their network you may be put on a blacklist, and access to the network will be blocked. The IMEI number can be found on the metal casing of the Quectel module.





## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

## 2 Firmware update

### 2.1 USB

For the update through the USB interface the tool Qflash is needed, this tool is included in the BG96 download\*.

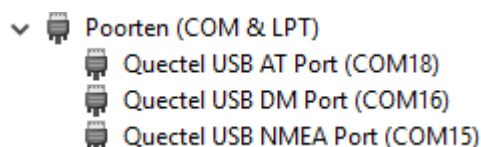
\*Please see chapter: [Set-up](#) for the download link.

In order to get a firmware package please contact Quectel or TOP-electronics via: [support@top-electronics.com](mailto:support@top-electronics.com).

Step 1: Turn on the ERF3000 shield by plugging in the adapter supplied with the shield and pressing the PWR button.

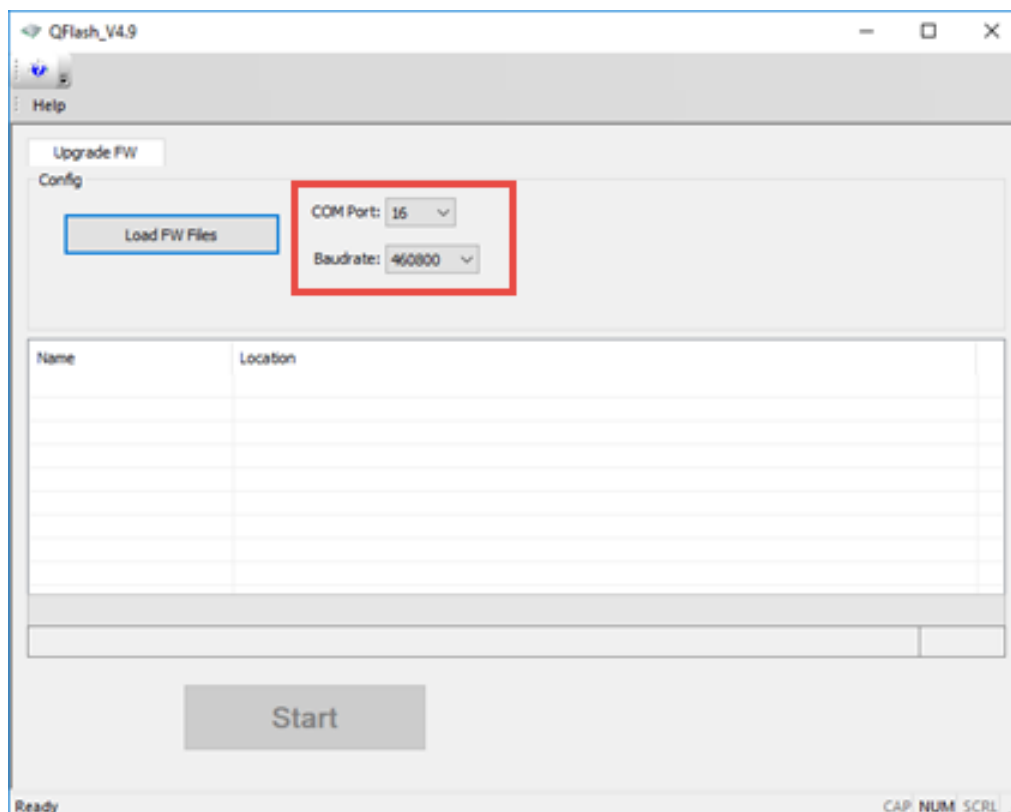
Step 2: Connect the ERF3000 to a laptop/PC through the USB interface.

Step 3: On your PC/Laptop look at your device management and find the "Quectel USB DM Port".



Step 4: Open the Qflash software

Step 5: Select the "Quectel USB DM Port" for the COM Port: and set the baudrate to 460800.







## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

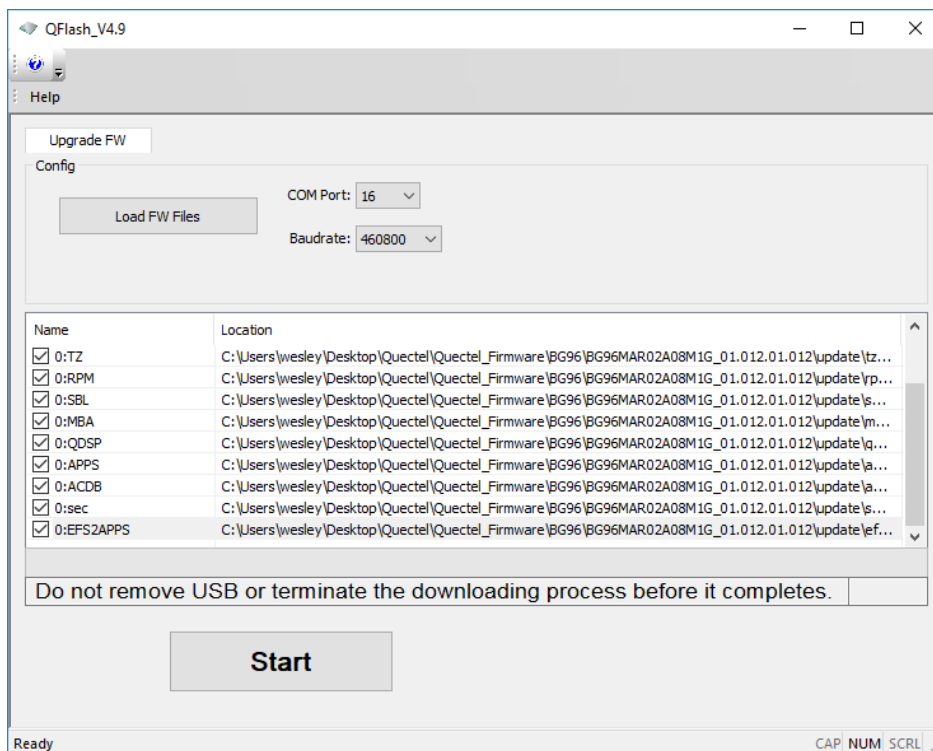
Step 6: Select the Firmware with the “Load FW Files”, This will open an explorer tab.

Navigate to the firmware folder and into the “update” folder.

BG96 > BG96MAR02A08M1G_01.012.01.012 > update				
Naam	Gewijzigd op	Type	Grootte	
acdb.mbn	12-3-2019 13:31	MBN-bestand	129 kB	
apps.mbn	12-3-2019 13:48	MBN-bestand	4.033 kB	
efs2apps.mbn	13-3-2019 04:09	MBN-bestand	24 kB	
ENPRG9x06.mbn	12-3-2019 13:06	MBN-bestand	96 kB	
mba.mbn	12-3-2019 13:50	MBN-bestand	241 kB	
NPRG9x06.mbn	12-3-2019 13:06	MBN-bestand	96 kB	
partition.mbn	12-3-2019 13:06	MBN-bestand	1 kB	
qdsp6sw.mbn	12-3-2019 15:06	MBN-bestand	23.023 kB	
rpm.mbn	12-3-2019 13:06	MBN-bestand	153 kB	
sbl1.mbn	12-3-2019 13:06	MBN-bestand	223 kB	
tz.mbn	12-3-2019 13:06	MBN-bestand	539 kB	

Select 1 of the “.mbn” files and click on “Open”. The software will automatically select all relevant files for the update.

Step 7: Click on the start button to begin the update.



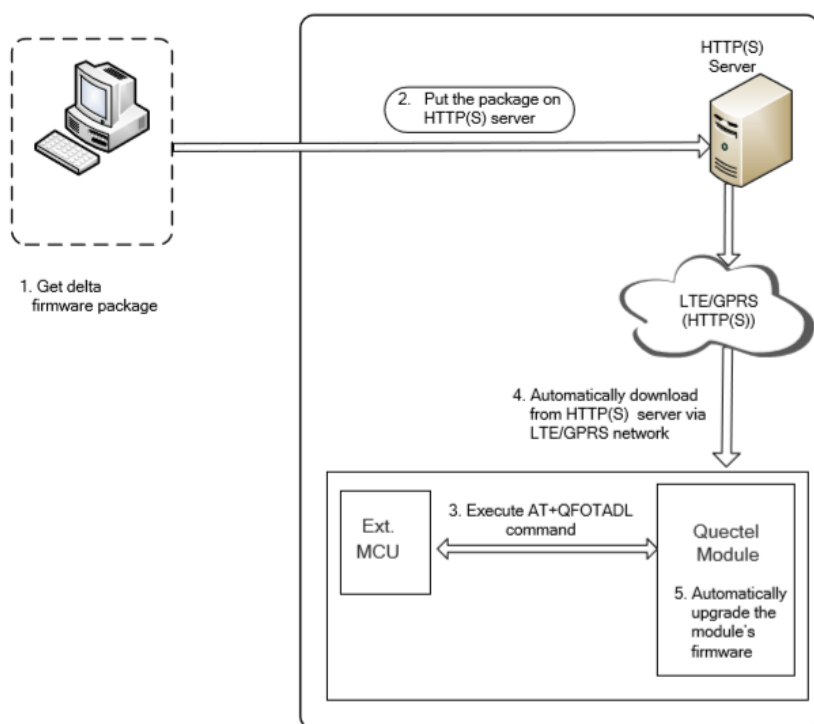


## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

### 2.2 DFOTA

In order to perform a DFOTA update, you will need to get a special DFOTA package. This can be obtained through contacting TOP-electronics via: [support@top-electronics.com](mailto:support@top-electronics.com) or Quectel. Please state your current firmware and specify what firmware version you would like to upgrade/downgrade to.

When the DFOTA package has been received please follow these steps:



Step 1: Get the delta firmware package.

Step 2: Put the delta firmware on a HTTP(s) server.

Step 3: Power on module and make sure a network connection is established.

Step 4: Execute AT+QFOTADL command. Then the module will automatically download the package from HTTP(S) server via LTE/GPRS network.

Command	Description
AT+QFOTADL=<httpURL>	This command will start the DOFTA process. (example AT+QFOTADL="https://www.quectel.com:100/update.zip")



## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

### 3 Creating a Low-Power application

#### 3.1 PSM

The Power saving mode (PSM) is the most important feature when creating a low-power application. This feature allows the module to enter a deep sleep mode for a specified time. When the specified time expires, the module will come back online. During the Power saving mode the current consumption drops to 10uA. The PSM can be configured with the following commands:

Command	Description
AT+QCFG="psm/enter",1	This command is used to trigger the module to enter into PSM mode immediately after the RRC released. When this function is enabled and RRC connection release is received, the module will skip active timer (T3324) and enter into PSM mode immediately
AT+QCFG="psm/urc",1	This command will allow the +QPSMTIMER URC message to be displayed, when the module receives the RRC release
AT+CPSMS=1,,,"TAU","active"	This command sets the TAU (T3312) and active (T3424) timers
AT+QPSMS=1,,,"TAU","active"	This command extends the previous set AT+CPSMS active and sleep timers

Please see chapter [5 Measuring the current consumption](#) for an example on how to measure the current consumption during this mode.

#### 3.2 eDRX

The extended discontinuous reception (eDRX) is a feature to lower current consumption while connected to a network. Using the eDRX settings the interval between communication to the network can be manually set. The default value for M2M communication is 10.24 seconds. For IoT applications this time can be prolonged to 2,9 hours. In between the eDRX paging cycles, the module will turn off it's radio and thus lowering the consumed current. Using the below commands the eDRX settings can be tuned.

Command	Description
AT+CEDRXS	This command will initialize the eDRX settings
AT+CEDRXRDP	This command will report the dynamic parameters of the extended eDRX settings

Please see chapter [5 Measuring the current consumption](#) for an example on how to measure the current consumption during this mode.

#### 3.3 Antenna design

The antenna design also has great influence on the total power consumption of the module. The consumed power is in direct correlation with the RSSI value. The RSSI value of the connection can be retrieved with the following command:

Command	Description
AT+CSQ	This command will return the signal strength (RSSI) of the connection

The RSSI value is in its place in correlation with the Impedance and the return loss of the antenna (design).

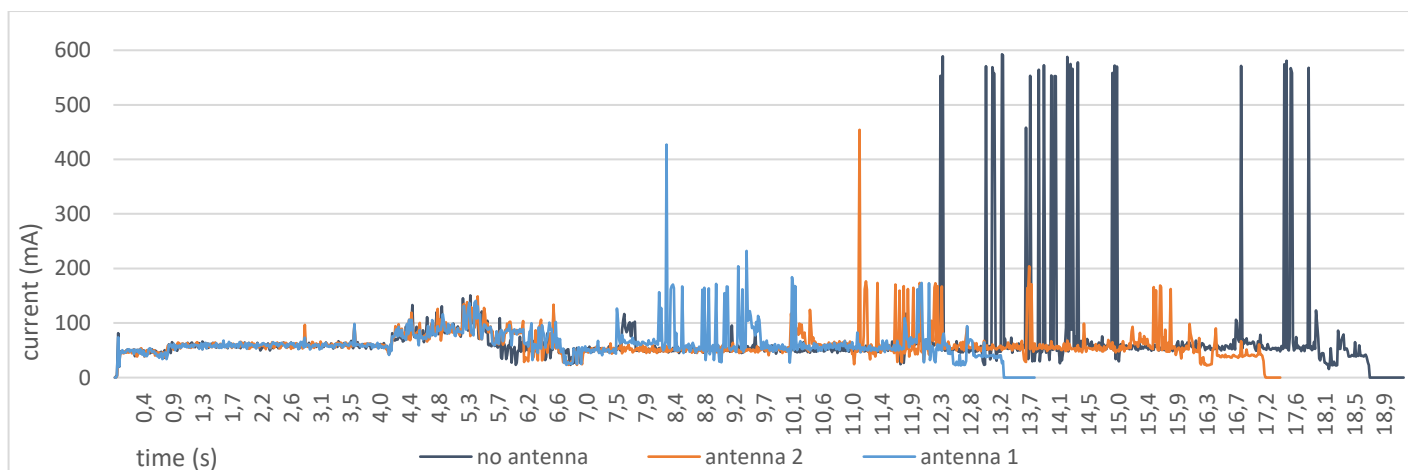


## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

In order to measure the return loss and impedance of an antenna (design), a Vector Network Analyzer (VNA) is needed. Using the VNA you can measure the impedance of your antenna design and get a representation of your antenna performance. To give an example of how important the antenna design is, see below test results:

The following tests are performed on the T-Mobile LTE Cat NB1 network. This network uses LTE B8 band with a center frequency of 900MHz.

Antenna	Return Loss @900 MHz	RSSI
Antenna 1 ERF4007	-16,14 dB	-59 dB
Antenna 2 ERF4041	-3,53 dB	-72 dB
No antenna	-0,22 dB	-106 dB



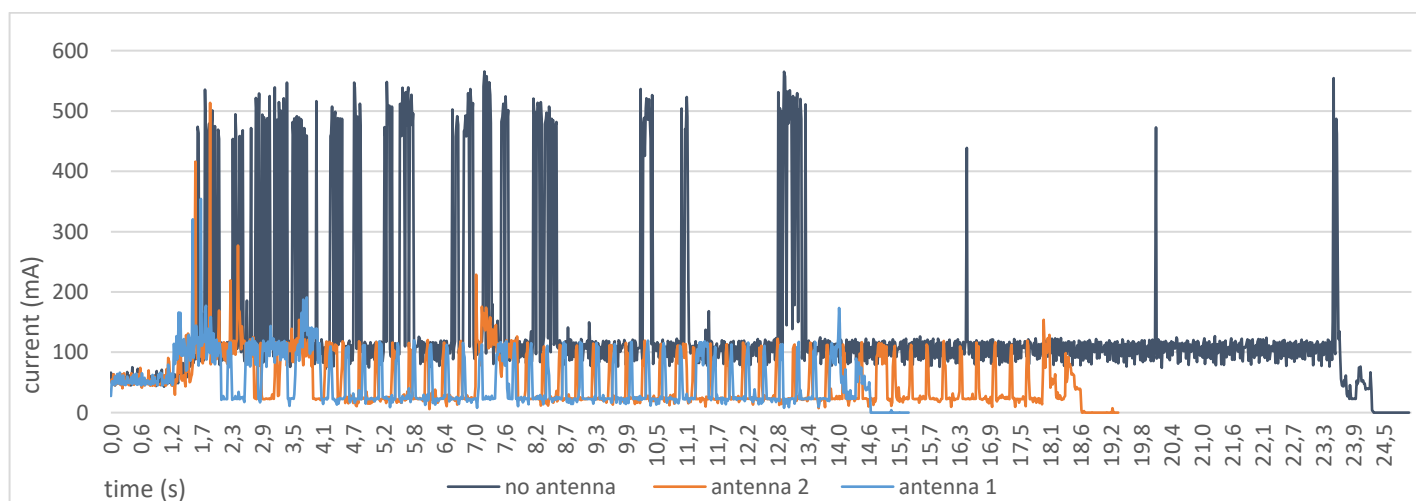
Results transmission	Avg current	Total time	Total	% increase in current
Antenna 1	62,26mA	13,75 sec	856,075mA	100%
Antenna 2	60,07mA	17,42 sec	1046,42mA	122%
No antenna	67,39mA	19,31 sec	1301,30mA	152%



## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

The following tests are performed on the KPN LTE Cat M1 network. This network uses LTE B20 band with a center frequency of 800MHz.

Antenna type	Return Loss @800 MHz	RSSI
Antenna 1 ERF4041	-13,13dB	-62dB
Antenna 2 ERF4061	-3.5 dB	-77
No antenna	-0.18dB	-101dB



Results transmission	Avg current	Total time	Total	% increase in current consumption
Antenna 1	50,25mA	15,34 sec	770,84mA	100%
Antenna 2	50,22mA	19,39 sec	973,77mA	126%
No antenna	144,74mA	24,98 sec	3615,61mA	369%

As seen in the graphs the antenna design is a big part of creating an optimal low power application. In order to get an optimal antenna performance, measurements with a VNA are necessary.

An example of a suitable VNA supplied by our partner TOP-electronics can be found here: <https://shop.top-electronics.eu/vna-starter-kit-0460e-p-18783.html>



## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

---

### 3.3.1 Antenna design tuning

In this subchapter the process of tuning an antenna design will be described. For all antenna measurements a VNA supplied by our partner TOP-electronics is used. More info about the VNA can be found here: <https://shop.top-electronics.eu/vna-0460e-400mhz-to-6ghz-2frac12-port-p-18787.html>

// antenne meting zonder filter

// antenne meting met filter

// performance verschil aantonen met en zonder filter -> koppelen aan energieverbruik



## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

---

### 4 Arduino Programming

#### 4.1 Setup

Start by downloading the Arduino IDE software on your PC. This can be found here:

<https://www.arduino.cc/en/Main/Software>. To communicate with the shield, a standard Arduino library called "SoftwareSerial" is needed. This is already installed with the Arduino IDE.

Information can be found here: <https://www.arduino.cc/en/Reference/SoftwareSerial>

Once Arduino IDE has finished installing open the software.

Connect the USB cable to the Arduino and to the PC. The green LED will light up, next press the PWR button. The Stat LED should light up and the Net light should start blinking.

#### 4.2 Example

//link to github?





## ERF3000v2, LTE Cat. M1, Cat.NB1, EGPRS & GNSS Arduino shield

---

### 5 Measuring the current consumption

#### 5.1 Preparations

//Removing resistor R...

+ image location

//soldering header on JP...

+ image location

#### 5.2 Results