µFR Mifare examples - Android
Version 1.0

Table of contents

Simplest **3**
Communication with µFR Reader 3
Reading card UID 4
Read and write operations 5

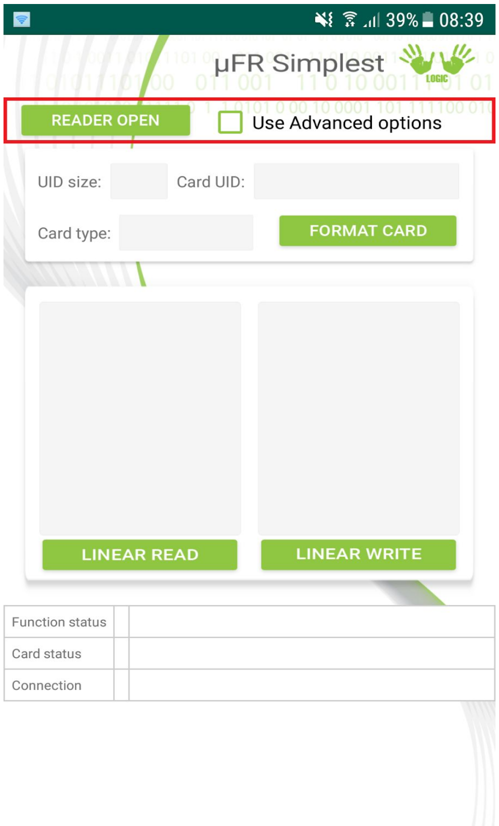
Simple **6**
Communication with µFR Reader 6
Reader options 7
Card options 8

Advanced **11**
Communication with µFR Reader 11
Reader options 11
List of options 12
Authentication 13
Block and Block in sector Read/Write 14
Value blocks 15

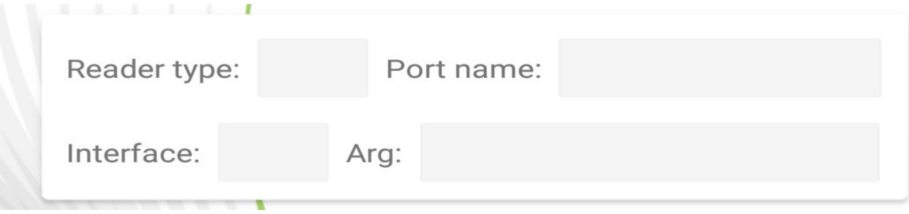
Revision history **20**

Simplest

Communication with μFR Reader

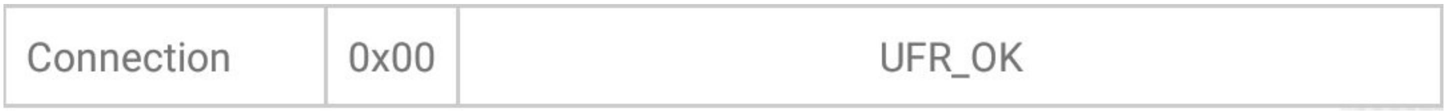


For opening communication with reader click **READER OPEN** button or select **Use Advanced options** if you are using for example μFR Online device, fill in the blanks and then click **READER OPEN**.



For **Reader type** fill empty field with **0**, for **Port name** fill empty field with reader **IP address**, for **Interface** fill empty field with **T** or **U** depending on your readers configuration, for **Arg** fill empty field with **0**.

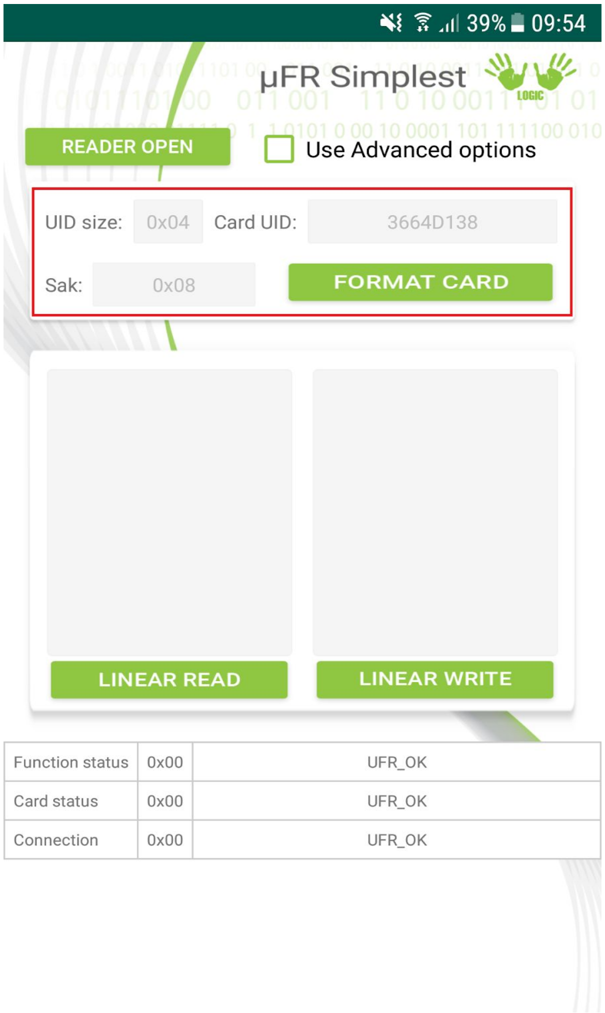
If establishing connection was successfull, you will see **UFR_OK** status next to **Connection**.



Connection	0x00	UFR_OK

Reading card UID

Every time you put your card on the reader you will be able to see its UID, Sak, and UID size.



□ Use Advanced options

UID size: 0x04 Card UID: 3664D138

Sak: 0x08

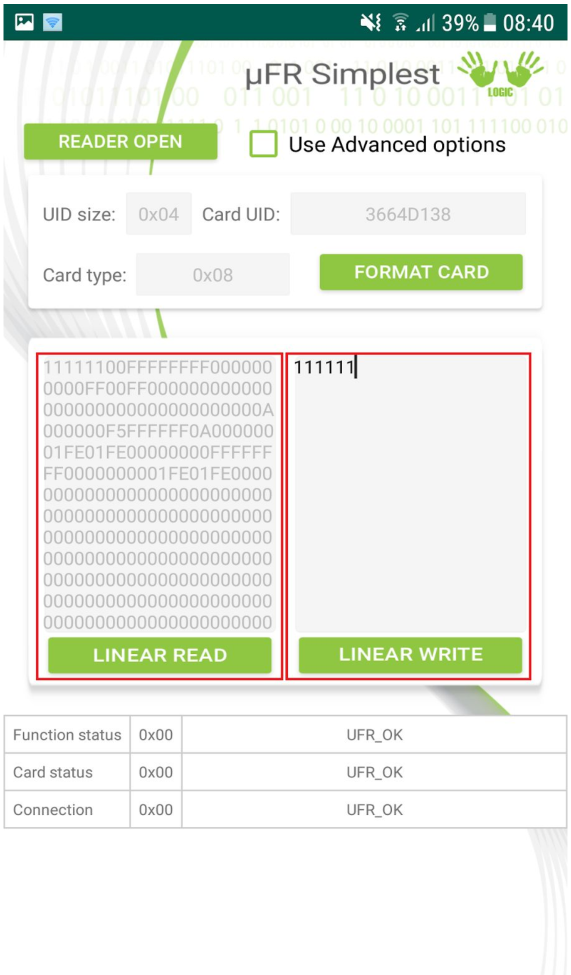
Function status	0x00	UFR_OK
Card status	0x00	UFR_OK
Connection	0x00	UFR_OK

If you click on **FORMAT CARD** button reader will write **Key A: 0xFFFFFFFFFFFF** and **Key B: 0xFFFFFFFFFFFF**

In all sectors on the card and will set **access bit 0** for all data blocks and **access bit 1** for all sector trailer blocks.

Read and write operations

For reading card, simply click on **LINEAR READ** button, and for writing, type data into text field in **hexadecimal format** and click **LINEAR WRITE** button.



□ Use Advanced options

UID size: 0x04 Card UID: 3664D138

Card type: 0x08

```
11111100FFFFFFFF000000
0000FF00FF000000000000
000000000000000000000A
000000F5FFFFFF0A000000
01FE01FE00000000FFFFFF
FF0000000001FE01FE0000
0000000000000000000000
0000000000000000000000
0000000000000000000000
0000000000000000000000
0000000000000000000000
0000000000000000000000
0000000000000000000000
```

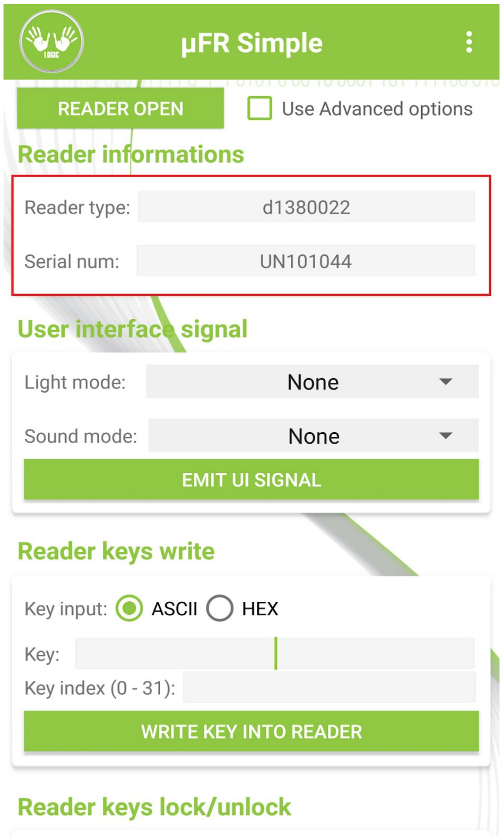
```
111111
```

Function status	0x00	UFR_OK
Card status	0x00	UFR_OK
Connection	0x00	UFR_OK

Simple

Communication with µFR Reader

Establishing communication with µFR Reader in Simple application is the same as in Simplest application.



µFR Simple

Reader type: d1380022

Serial num: UN101044

Light mode: None

Sound mode: None

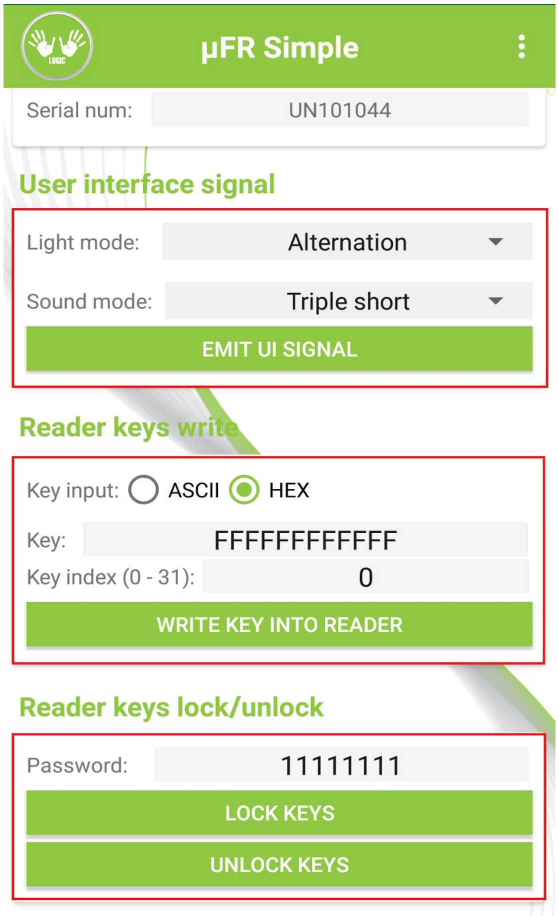
Key input: ● ASCII ○ HEX

Key:

Key index (0 - 31):

If establishing communication was successfull, you will be able to see readers type and serial number as shown in the picture above.

Reader options



µFR Simple

Serial num: UN101044

User interface signal

Light mode: Alternation

Sound mode: Triple short

Reader keys write

Key input: ○ ASCII ● HEX

Key: FFFFFFFFFFFF

Key index (0 - 31): 0

Reader keys lock/unlock

Password: 11111111

- User interface signal

For emmiting user interface signal, choose wanted **light and sound** mode from dropdown menu and click **EMIT UI SIGNAL** and you will be able to hear beeping and to see readers light signal.

- Reader key write

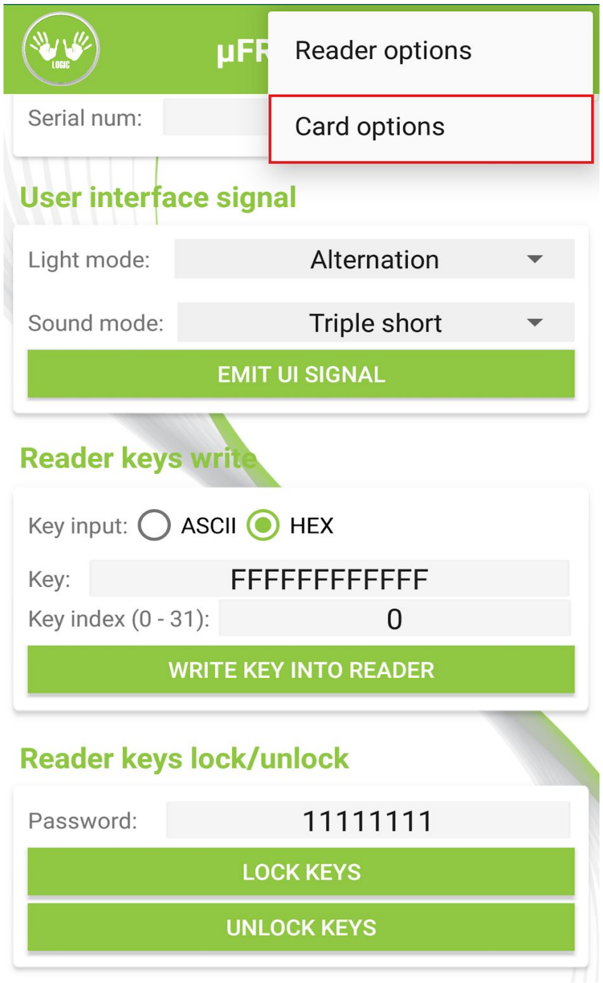
For writing key into reader, first, select **key input mode** (ASCII or HEX) then type in **key** you want to store into empty text field and type in **key index** for storing into reader, at the end, click **WRITE KEY INTO READER.**

- Reader key lock/unlock

This option allows you to lock or unlock reader for writing keys into it. Type in **8 characters long password** and click **LOCK KEYS** or **UNLOCK KEYS** button.

Card options

For opening Card options, click on three white dots at the top right corner and select Card options



Serial num:

User interface signal

Light mode: Alternation

Sound mode: Triple short

Reader keys write

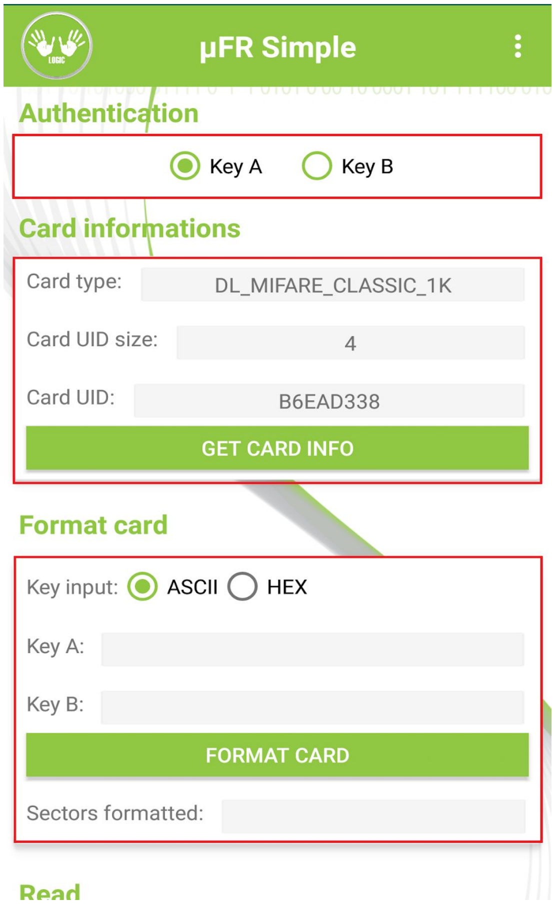
Key input: ○ ASCII ● HEX

Key: FFFFFFFFFFFF

Key index (0 - 31): 0

Reader keys lock/unlock

Password: 11111111



Authentication

● Key A ○ Key B

Card informations

Card type: DL_MIFARE_CLASSIC_1K

Card UID size: 4

Card UID: B6EAD338

Format card

Key input: ● ASCII ○ HEX

Key A:

Key B:

Sectors formatted:

Read

- Authentication

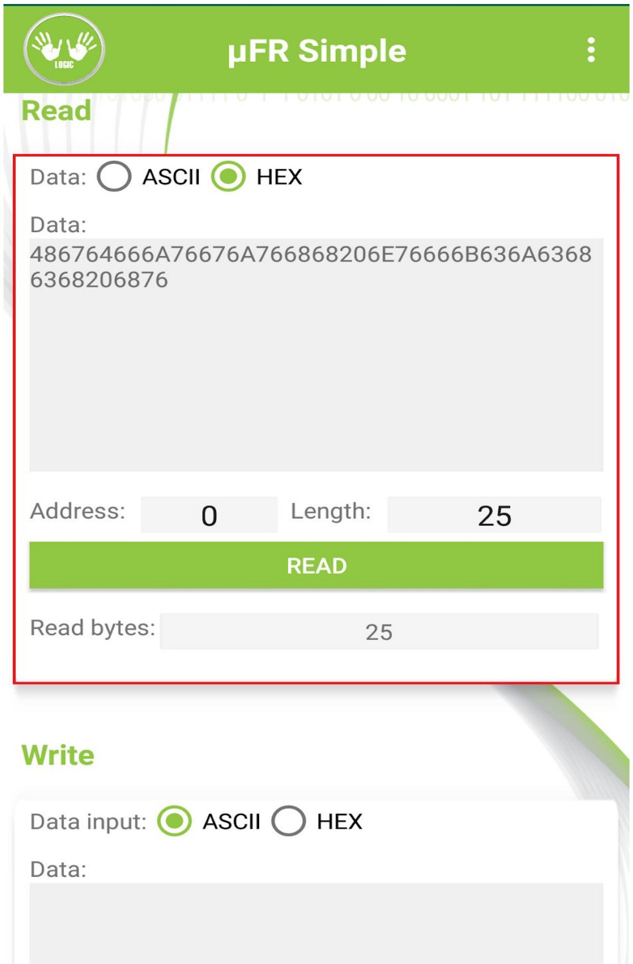
In this section you can switch authentication from **key A** to **key B** and vice versa.

- Card informations

Put card on the reader and simply click on **GET CARD INFO** button to see **cards type**, **UID** and **UID size**.

- Format card

You can use format card option to store **new key A** and **new key B** into all sectors in card.



Read

Data: ○ ASCII ● HEX

Data:
486764666A76676A766868206E76666B636A6368
6368206876

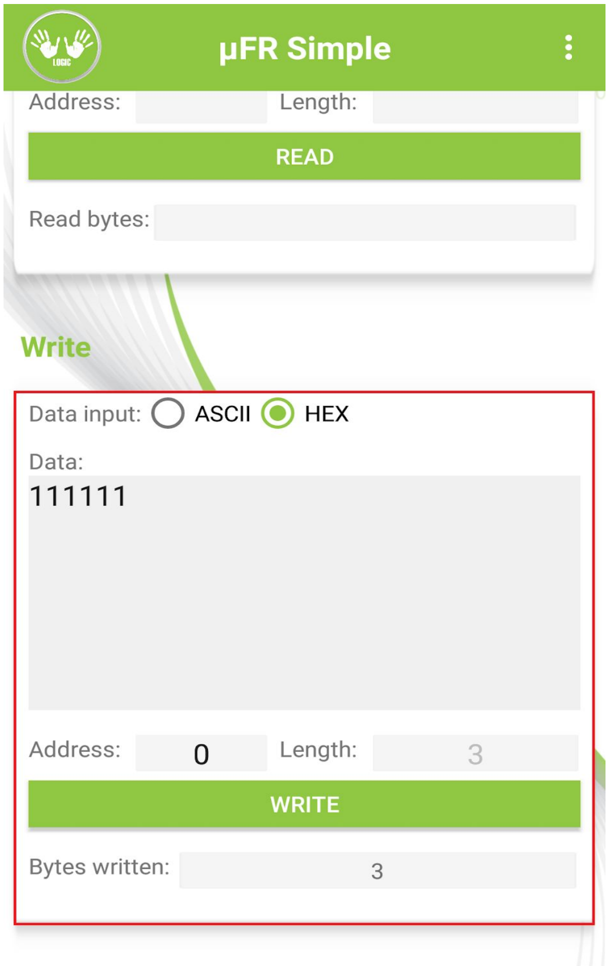
Address: 0 Length: 25

Read bytes: 25

Write

Data input: ● ASCII ○ HEX

Data:



Address: Length:

Read bytes:

Write

Data input: ○ ASCII ● HEX

Data:
111111

Address: 0 Length: 3

Bytes written: 3

- Read

For reading, type in **address** (where to start reading) and how many bytes to read (into empty **length** field) and simply click **READ** button to see card data. Also, you can choose data representation switching between ASCII and HEX radio buttons.

- Write

For writing, select what kind of data you want to store (**ASCII** string or **HEX** string), type in data into empty text field, type in **address** (where to start writing) and simply click **WRITE** button, length will be automatically calculated.

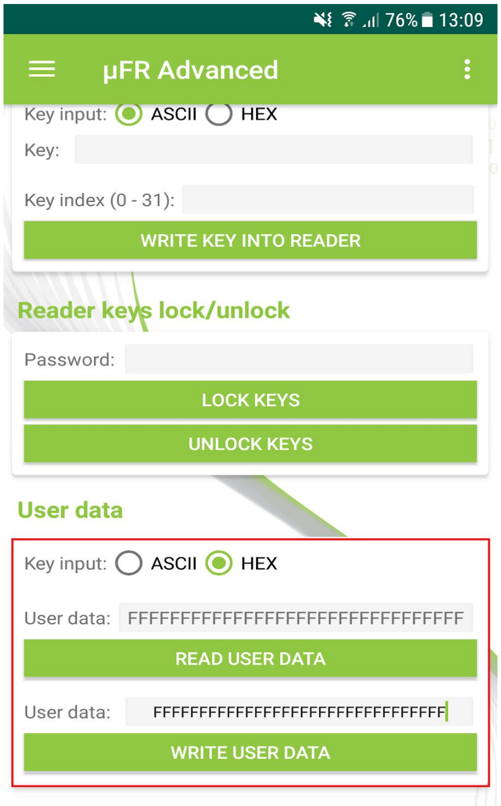
Advanced

Communication with µFR Reader

Establishing communication with µFR Reader in Advanced application is the same as in Simplest application.

Reader options

All reader options are mostly the same as in the Simple application except one extra option named **User data**. In User data option you will be able to read and write 16 bytes from reader EEPROM if you click on button **WRITE USER DATA**. See picture below:



Key input: ● ASCII ○ HEX

Key:

Key index (0 - 31):

Password:

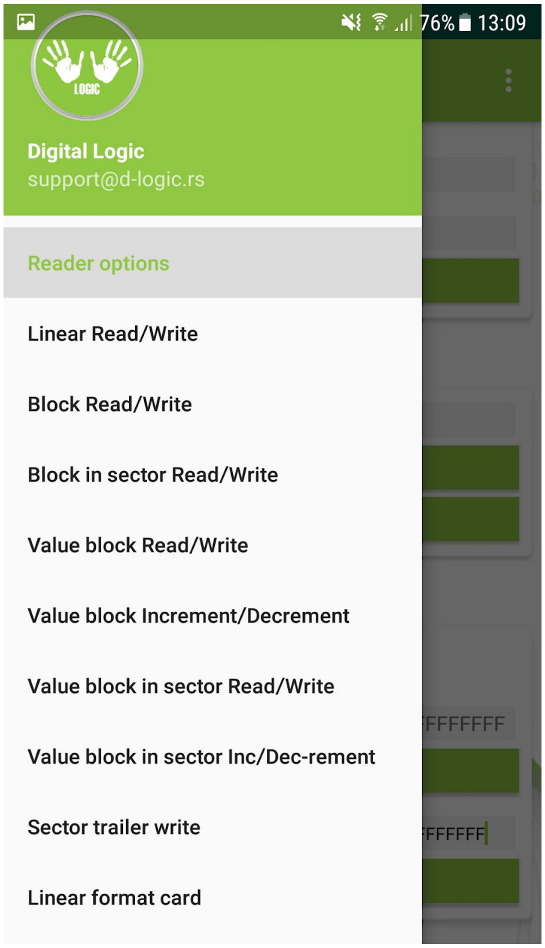
Key input: ○ ASCII ● HEX

User data: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

User data: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

List of options

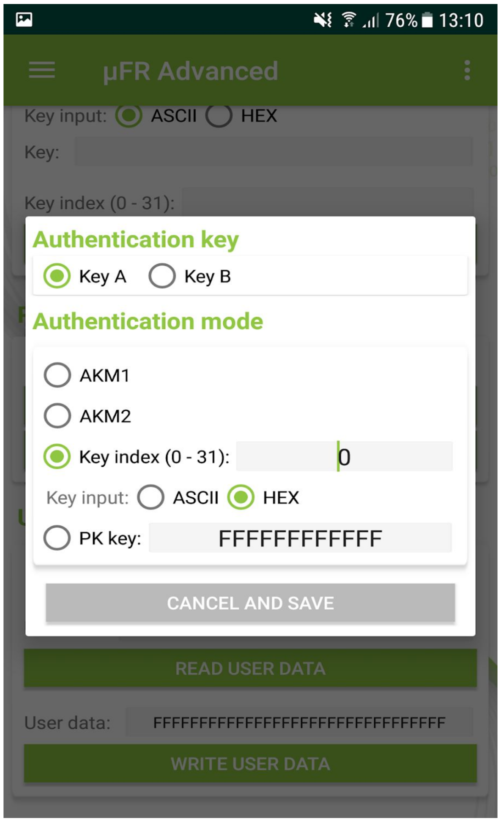
In Advanced application, there are multiple ways to read and write data from card, such as:



- Linear Read/Write - works the same as in the Simple application.
- Block and block in sector Read/Write
- Value block and value block in sector Read/Write/Increment/Decrement
- Sector trailer write
- Linear format card

Authentication

For authentication settings, click on three white dots in top right corner and select Authentication settings.



76% 13:10

µFR Advanced

Key input: ● ASCII ○ HEX

Key:

Key index (0 - 31):

Authentication key

● Key A ○ Key B

Authentication mode

○ AKM1

○ AKM2

● Key index (0 - 31): 0

Key input: ○ ASCII ● HEX

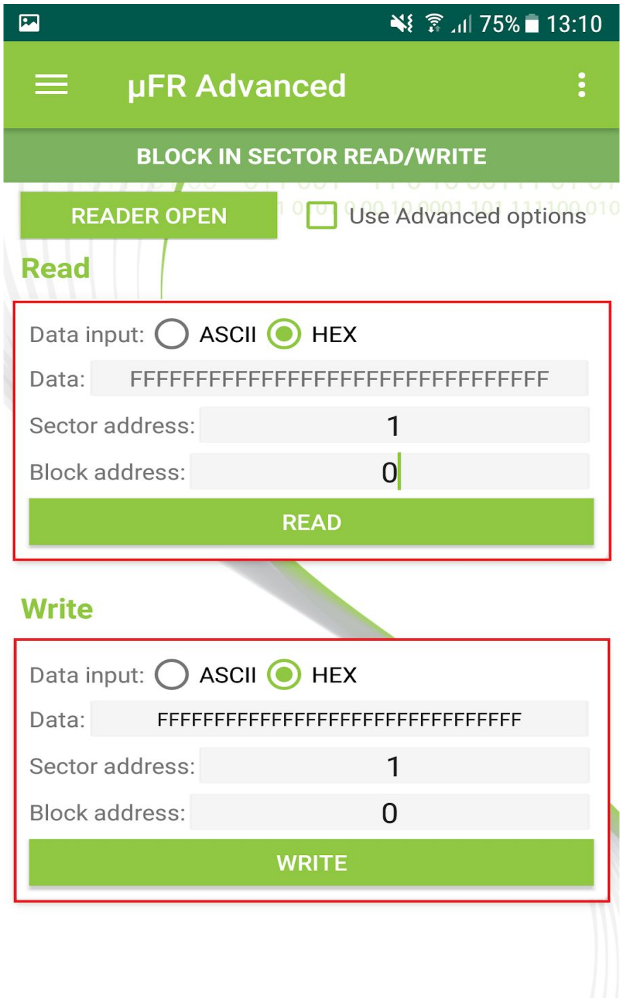
○ PK key: FFFFFFFFFFFF

User data: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

In this pop-up window you will be able to switch authentication key from A to B and vice versa.
Also you can switch between authentication modes: Automatic key mode 1, Automatic key mode 2, Key index and Provided key. For explanations about theese authentication refer to µFR Series NFC Reader API.

Block and Block in sector Read/Write

Theese two options are very similar, the only difference is in Block in sector read and write options you have to provide both, sector and block address and in block read and write options, you have to provide only block address for read and write operations. Here is some example of block in sector reading and writing data:



□ Use Advanced options

Read

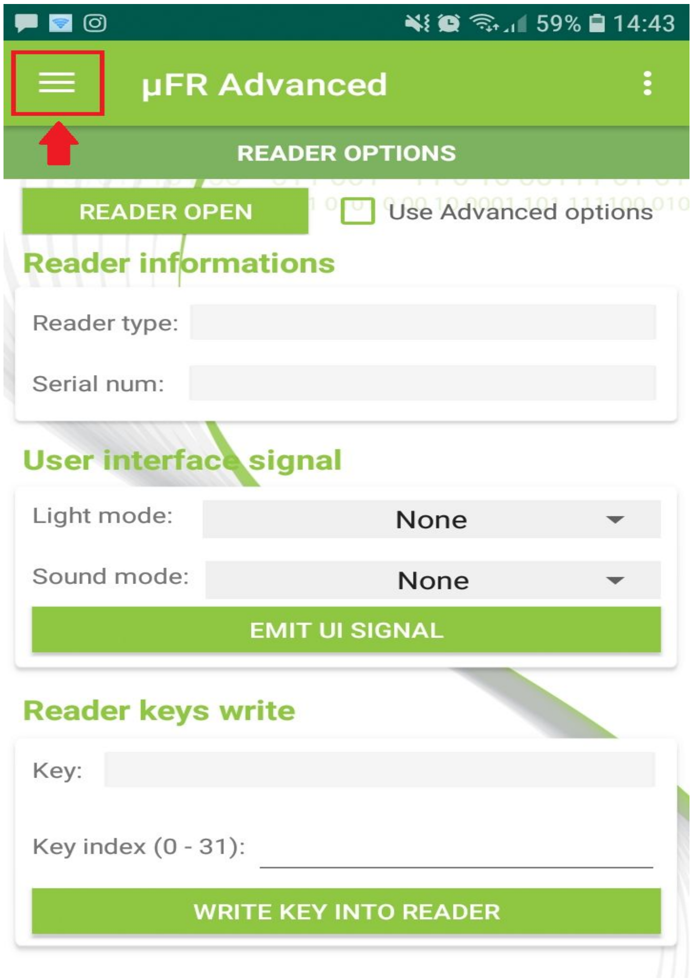
Data input: ○ ASCII ● HEX
Data: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Sector address: 1
Block address: 0

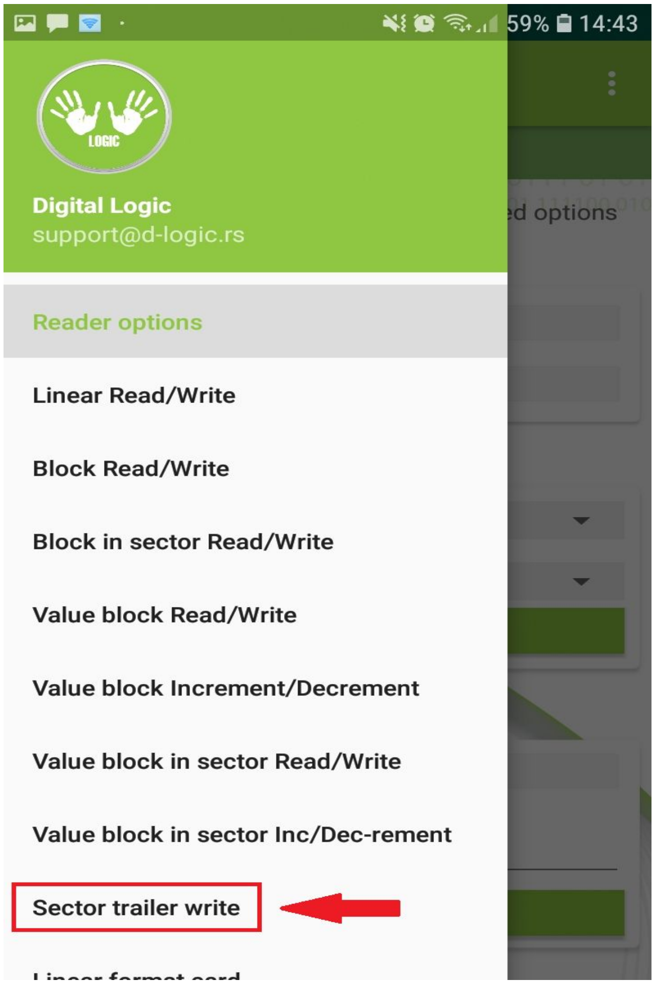
Write

Data input: ○ ASCII ● HEX
Data: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Sector address: 1
Block address: 0

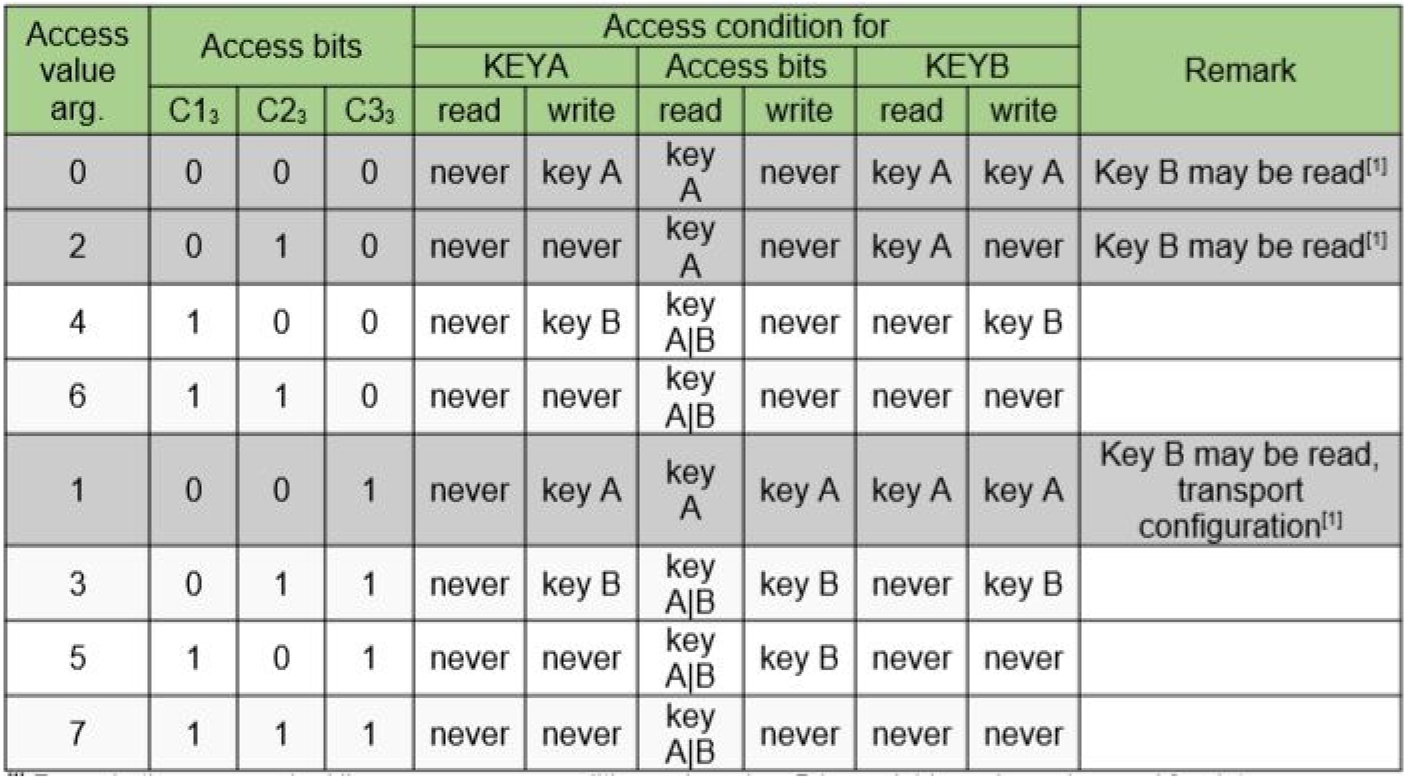
Value blocks

If you want to configure blocks as value blocks, you have to change blocks access bits. Click on side bar menu and then select 'Sector trailer write'' option.



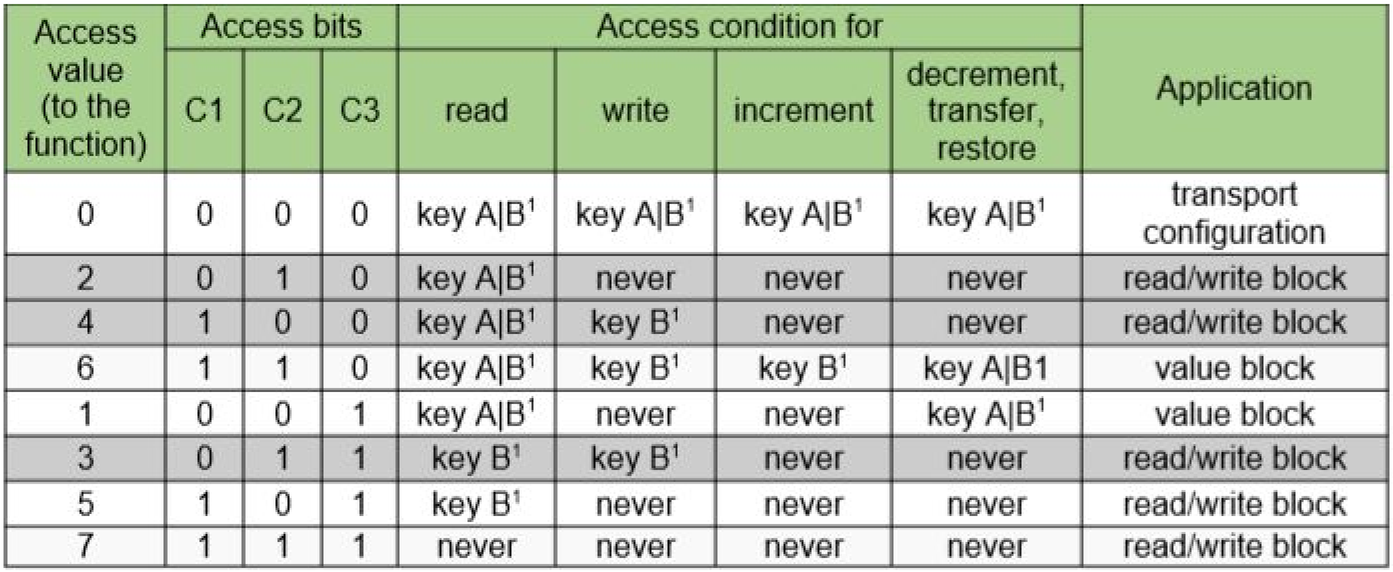


Sector trailer access bits:



Access value arg.	Access bits			Access condition for						Remark
				KEYA		Access bits		KEYB		
	$C1_3$	$C2_3$	$C3_3$	read	write	read	write	read	write	
0	0	0	0	never	key A	key A	never	key A	key A	Key B may be read[1]
2	0	1	0	never	never	key A	never	key A	never	Key B may be read[1]
4	1	0	0	never	key B	key A|B	never	never	key B	
6	1	1	0	never	never	key A|B	never	never	never	
1	0	0	1	never	key A	key A	key A	key A	key A	Key B may be read, transport configuration[1]
3	0	1	1	never	key B	key A|B	key B	never	key B	
5	1	0	1	never	never	key A|B	key B	never	never	
7	1	1	1	never	never	key A|B	never	never	never	

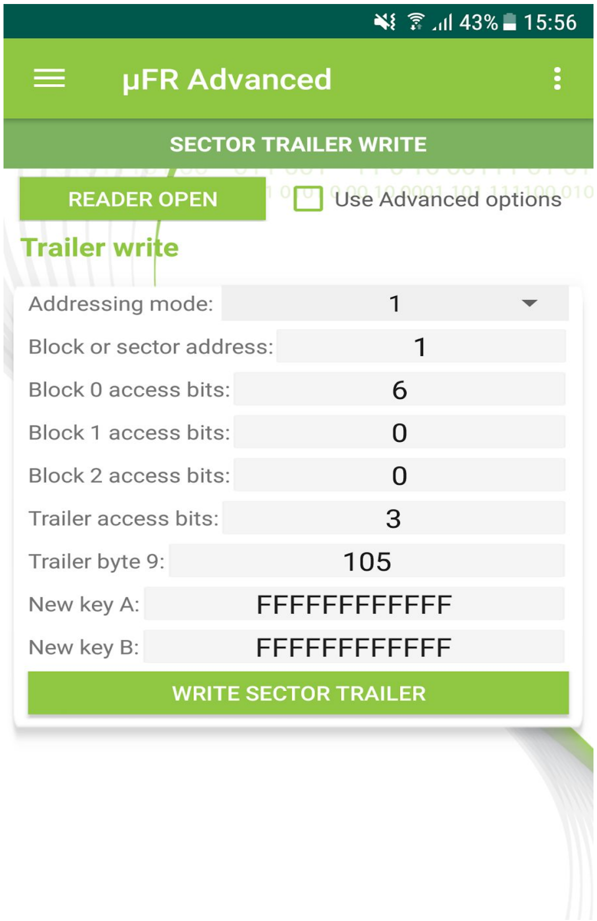
Block access bits:



Access value (to the function)	Access bits			Access condition for				Application
	C1	C2	C3	read	write	increment	decrement, transfer, restore	
0	0	0	0	key A|B^1	key A|B^1	key A|B^1	key A|B^1	transport configuration
2	0	1	0	key A|B^1	never	never	never	read/write block
4	1	0	0	key A|B^1	key B^1	never	never	read/write block
6	1	1	0	key A|B^1	key B^1	key B^1	key A|B1	value block
1	0	0	1	key A|B^1	never	never	key A|B^1	value block
3	0	1	1	key B^1	key B^1	never	never	read/write block
5	1	0	1	key B^1	never	never	never	read/write block
7	1	1	1	never	never	never	never	read/write block

For configuring data blocks as value blocks, please refer to [μFR Series NFC Reader API]

For example, we will configure first block in first sector as 'VALUE' block.



Addressing mode: 1

Block or sector address: 1

Block 0 access bits: 6

Block 1 access bits: 0

Block 2 access bits: 0

Trailer access bits: 3

Trailer byte 9: 105

New key A: FFFFFFFFFFFF

New key B: FFFFFFFFFFFF

<u>Addressing mode:</u>

0 - absolute sector trailer address

1 - relative sector trailer address

<u>Block or sector address:</u>

It depends on your addressing mode choice

<u>Block 0 access bit:</u>

We will set '6' for value block

<u>Block 1 access bit:</u>

We will set '0' and block 1 will stay 'data block'

<u>Block 2 access bit:</u>

We will set '0' and block 2 will stay 'data block'

<u>Sector trailer access bit:</u>

We will set '3'

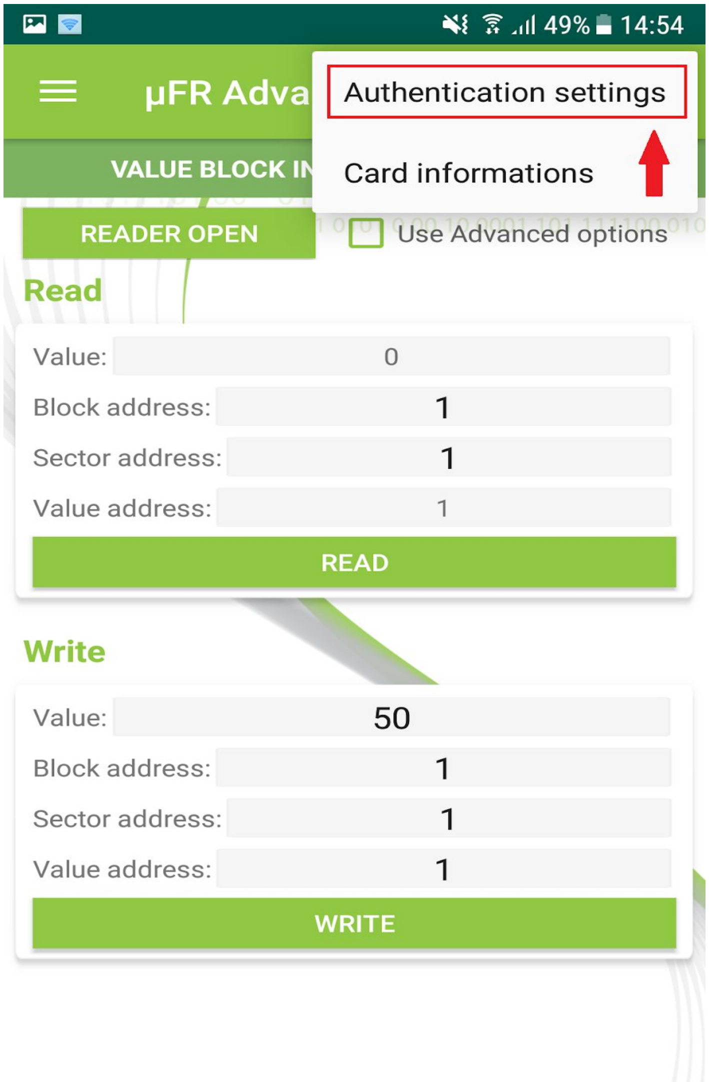
<u>Sector trailer byte 9:</u>

We will set default value 105 (0x69 hexadecimal)

We will use default key A and B (0xFFFFFFFFFFFF)

With this access bits configuration, you can read and decrement value blocks with both keys (A or B), but for writing and increment valuse you have to use key B. For switching authentication key to key B, follow these steps:

1. Click on three white dots at the top right corner
2. Choose 'Authentication settings option
3. Select 'Key B' as authentication key



□ Use Advanced options

Read

Value: 0

Block address: 1

Sector address: 1

Value address: 1

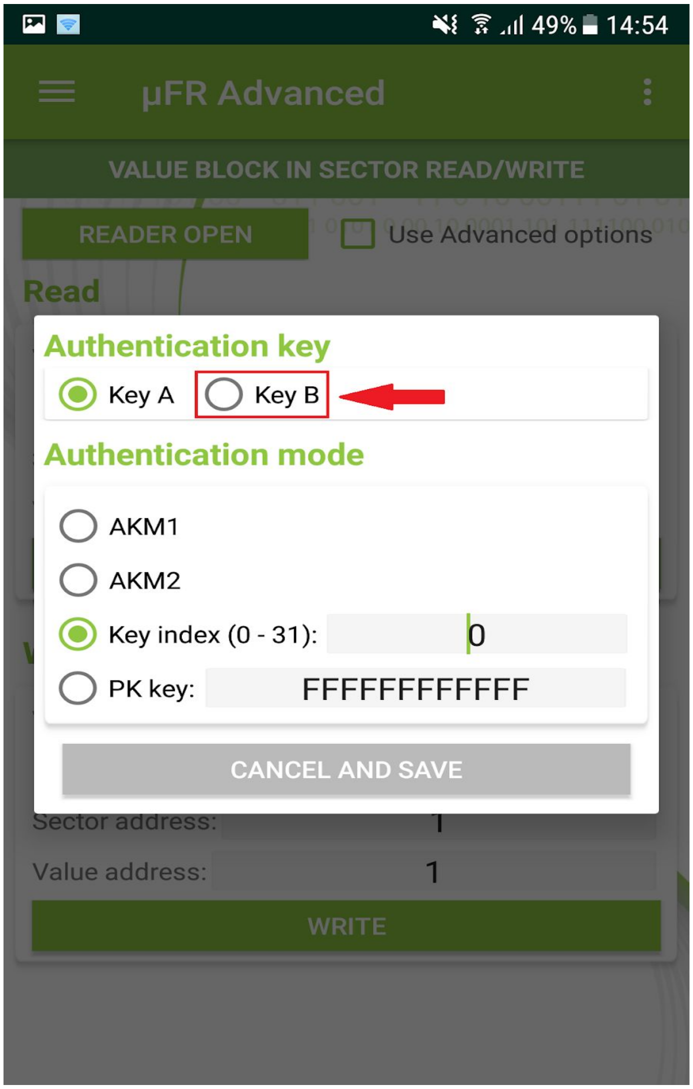
Write

Value: 50

Block address: 1

Sector address: 1

Value address: 1



□ Use Advanced options

Authentication key

● Key A ○ Key B

Authentication mode

○ AKM1

○ AKM2

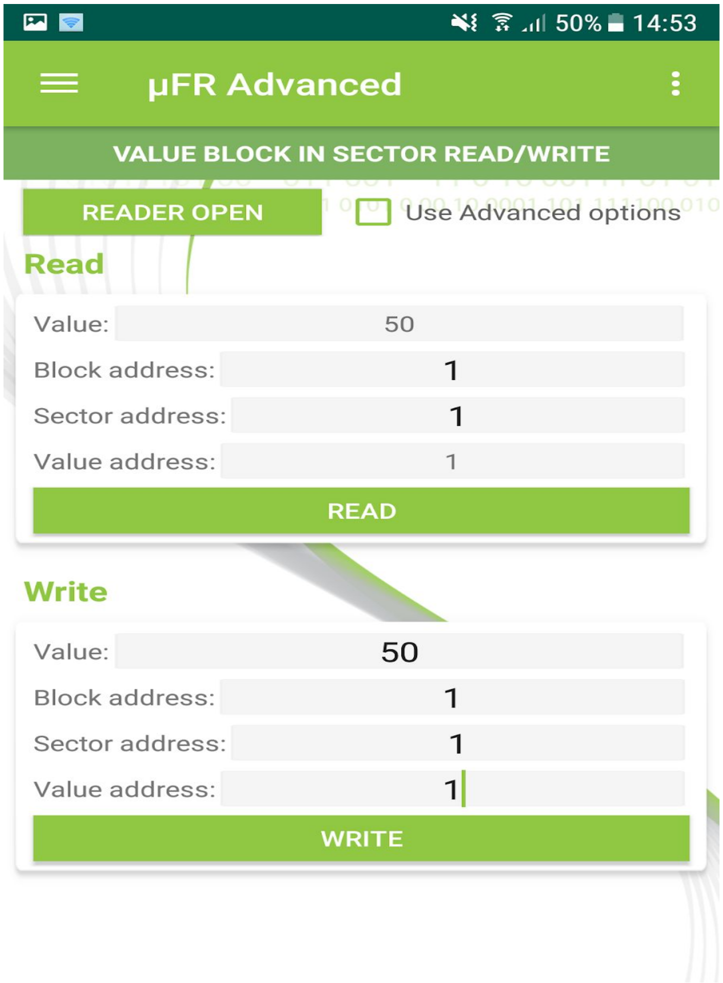
● Key index (0 - 31): 0

○ PK key: FFFFFFFFFFFF

Sector address: 1

Value address: 1

After successfull configuration of value blocks you can, for example, choose ‘Value block in sector Read/Write’ or ‘Value block in sector Increment/Decrement’ option in sidebar menu:



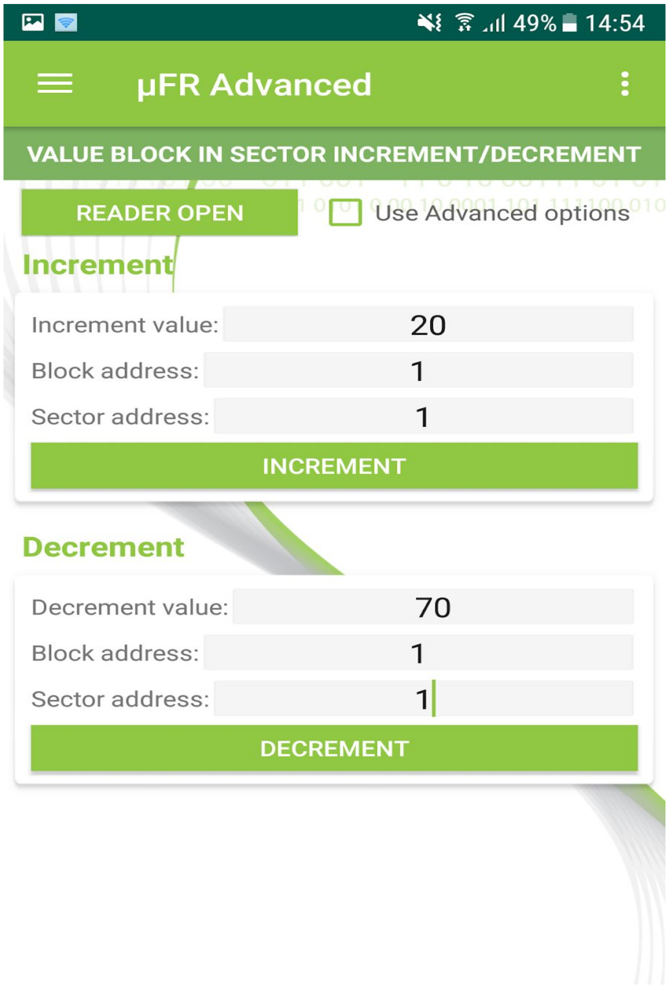
☐ Use Advanced options

Read

Value: 50
Block address: 1
Sector address: 1
Value address: 1

Write

Value: 50
Block address: 1
Sector address: 1
Value address: 1



☐ Use Advanced options

Increment

Increment value: 20
Block address: 1
Sector address: 1

Decrement

Decrement value: 70
Block address: 1
Sector address: 1

Revision history

Date	Version	Comment
2019-20-09	1.0	Base document