

Setting up the Bluetooth remote control

R1 Mini Ring / MagicSee

for use
with LocusMap 4.xx

The following document describes how to set up a Bluetooth remote control "R1 Mini Ring" -also called "MagicSee"- for use as a remote control for the "Locus Map 4.x" app (hereafter: LM) in connection with any smartphone.

This is particularly interesting for cyclists, but also for motorcyclists.

The remote control (hereafter: RC) allows you to operate the essential elements "*while driving*". This means you no longer have to stop or take your hands off the handlebars and try to operate the smartphone's touchscreen while driving: this is difficult and dangerous!!!

You can find the RC at many suppliers on the web and it costs about 10\$. It has an integrated battery that lasts a long time. To charge it, the RC is removed from the bike and charged with a micro USB charger or you leave it on the bike and charge it with a powerbank (red LED turns on). The charging process is usually completed in a few minutes (red LED is off); in my experience, the battery lasts "*several days*" with various bike tours.

This document does not describe the necessary conversion of the RC for mounting on the bicycle/motorcycle handlebars. This is of course absolutely necessary, because you want to operate the RC with your thumb while gripping the handlebars. Search the forum on this topic and you will get some ideas.

Here below is a *picture of the RC "R1 / MagicSee"* with the existing buttons:



To **set up the RC with LM**, a few **settings** must first be made, the most important of which is the assignment of so-called "**codes**" to the individual buttons of the RC in LM. To do this, however, you must first know these "**codes**".

In the case of the forum expert "*freischneider*" and myself, the codes were identical, so one could assume that they should also be the same for „your" R1/MagicSee" RC.

If you want to continue with these "**codes**" for now, which will probably work, **go directly to step 2**.

This is definitely much faster and less time-consuming, so you should definitely try it first.

If it doesn't work, you can always continue with **step 1** in the second attempt.

Step 1 - Reading out the button codes on the RC

(only necessary if steps 2 to 4 did not work in your first attempt).

The RC has hard-programmed "**codes**" for each individual button. The first thing we have to do is find out what these **codes** are. To do this, you need the app "**KeyEventDisplay**". I no longer found it in the *Google Playstore* and am attaching it here as an *.apk installation file. If you can't find this file here for copyright reasons, Google it. It is freely available for download on the web, e.g. here: <https://m.apkpure.com/de/keyevent-display/aws.apps.keyeventdisplay>

The app must then first be successfully installed.

After installing the "**KeyEventDisplay**" app, the RC is switched on and connected to the smartphone via Bluetooth (BT).

Then a so-called "**preset**" must be selected.

NOTE: The six (operating) buttons of the RC and the joystick can be runned under four different "**presets**" (A-D). This gives the buttons more functions at the same time, as you can also change the **presets**.

For our use here with LM, this is rather uninteresting and I will only describe **the setting & use of "Preset B"** below.

After the RC has been switched on and connected to the phone, we now activate "**Preset B**". This is done by "**pressing & holding**" the "**M**" **button** and then short pressing the "**B**" **button** at the same time. The blue LED on the RC lights up shortly and then goes off again.

Preset B is now activated.



Now start the app "**KeyEventDisplay**" on the smartphone. It appears with a spartan, almost completely black screen (see left picture).

Now you have to press all six buttons of the RC, one after the other, and write down the "**codes**" determined by the app.

For example, you press the "**A**" **button** on the RC. Immediately after the key is pressed, two lines appear in the app for **each pressed button**.

We are not interested in the whole cryptic string, but only in one or two numbers that are located relatively at the beginning of the lines (*circled in yellow*).

These numbers can be *the same or different* in both rows.

In the *picture on the right*, you can see that for the first **button** (in lines 1 and 2), both times the same number "96" was determined as the **code**.

The next but one **button** then produced the different **codes** "99" and "104" in lines 5 and 6.

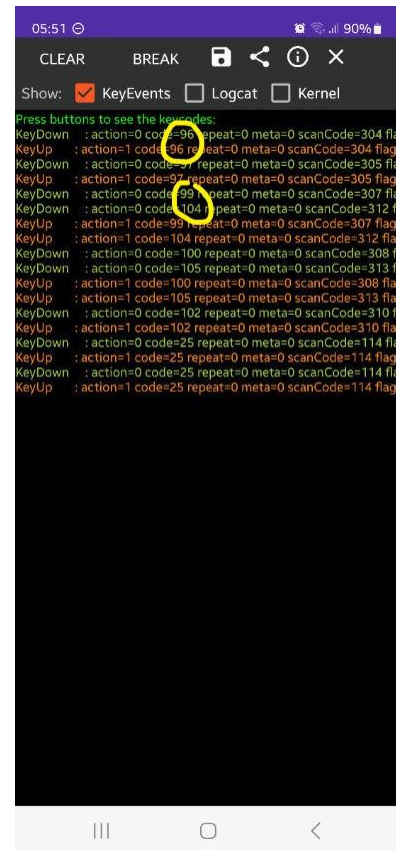
Now do this for all six (operating) **buttons** of the RC, i.e. key A-D, end up, & end down).

We don't need the joystick here.

The **"M"** and **"On/Off"** **buttons** are taboo and are not pressed!

You can also press a button several times if you are no longer sure which codes belong to which button. The app is patient and always throws out two lines (again) per keystroke....

At the end of the "reading" process, you will have received one or two numbers as a **"code"** for each button.



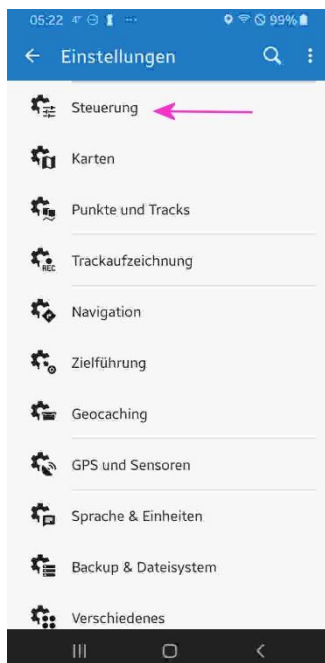
Where you have received two different codes for one button, you will have to try them both later in **step 3** when **assigning the codes in LM**.

The app has done its job and can be closed.

Step 2 - The basic settings in Locus Map (LM)

To use the RC with the app "Locus Map", we must first make a few basic settings in LM.

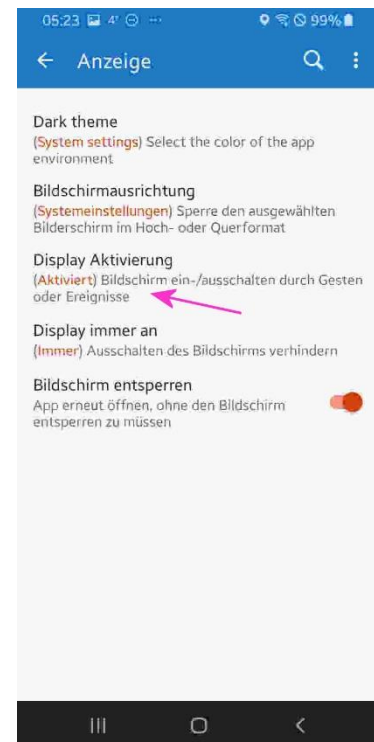
To do this, we call up the **"Settings"** of LM and go to the top menu sub-item, the **"Settings"** (picture on the left).



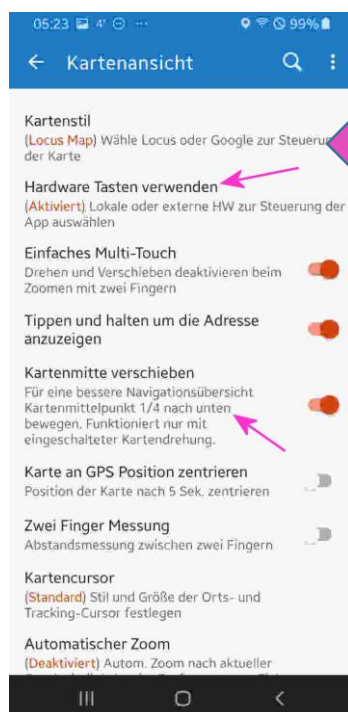
In the submenu that appears, we call up the menu item "**Display**" and make the following settings (picture on the right).

NOTE: You have to experiment with the menu item "**Display Activation**". At the beginning I had set the function to "**activated**", but then the screen came on '**suddenly**' too often when driving (I didn't know why) and after I had "**deactivated**" this function, it became better. I would like to switch off the screen more often when navigating to save battery power - and I don't want it *to switch on again automatically!*

So try it out...☺



We go back one menu level again and select the menu sub-item "**Map View**" and make the following settings: (picture below left):



NOTE 1: the item "**Use hardware buttons**" must **UNCONDITIONALLY** be set to "**activated**". Otherwise your RC will not run as desired despite all efforts (thanks again, *freischneider*).

NOTE 2

The item "**Move map centre**" does not necessarily have to be activated, but in my opinion it makes sense.

This function has the effect that if the map view in LM is later set to "**rotating map**" (i.e. the path in front of you is "**always at the top**"; you also know this from the Auto-Navi), the cursor with your position is not exactly in the middle of the screen, but a little further down. The advantage is that you now see "**more**" of the path in front of you on the screen.

However, this function only plays a role if "**rotating map**" is activated. With the *classic* orientation of the map (north always on top), the cursor with your position in LM is always in the centre of the screen.

We go back one menu level again and have now made the general settings for the integration of the RC.

In the sub-item "**Control bars & buttons**" you can make many more settings, but not all of them can be illuminated here. However, this has no effect on the functioning of the RC.

Step 3 - Assigning the (key) codes in Locus Map

Now comes the crucial point, because we now assign the (key) **codes** to the functions in "Locus Map".

You have the **codes** of the six RC keys either

- a) Determined in [step 1](#) with the app "**KeyEventDisplay**" or
- b) you take them from the following table (column B)

Keycodes:

Presets

Mode	A	B	C	D
↶ (end side upper) <i>rotate map</i>	97	<u>103</u>	103	4
○ (end side lower) <i>screen off/on</i>	96	<u>102</u>	102	---
Joystick up	24	19	19	mouse
Joystick down	25	20	20	mouse
Joystick left	88	21	21	mouse
Joystick right	87	22	22	mouse
A <i>not used</i>	85	100,105	97	96
B <i>Zoom in</i>	---	<u>96</u>	99	97
C <i>center map</i>	25	<u>99,104</u>	96	24
D <i>Zoom out</i>	24	<u>97</u>	100	25

Since we will only focus on "**Preset B**" (values in **column B**, **outlined in red**) in the following, I will not refer to the other columns any more.

So on the *left in the table* is my **preset B**:

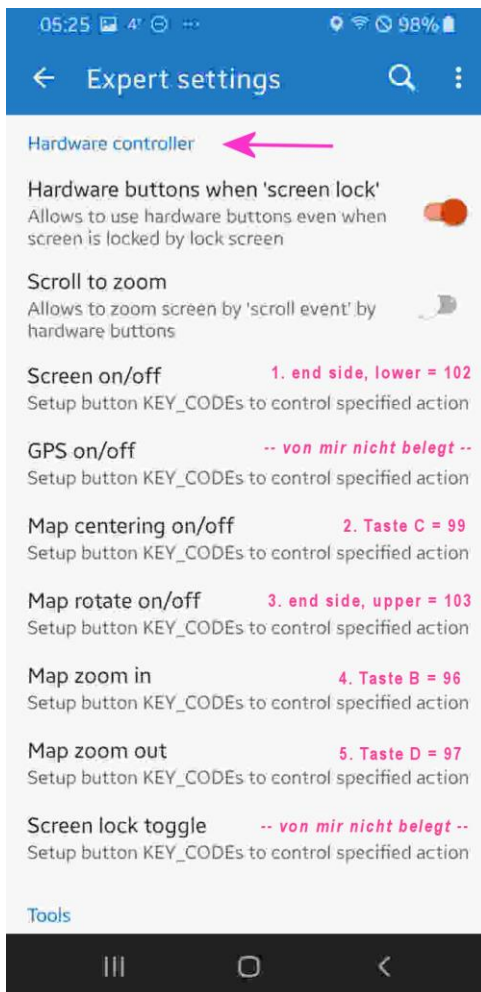
I used the **codes** underlined in green in **column B** in LM

The above codes in the table are not working for you?

➡ Then go back to [step 1](#) - Reading out the key codes on the RC

NOTE: IN THE table you will find the key codes of all the keys on my RC.

These codes were also used by the forum expert "*freischneider*". So with a bit of luck they are also your codes! But since no one knows whether all "R1/MagicSee" RCs are soldered together in one and the same shop in China, it is possible that other codes were programmed into your RC....in which case only [step 1](#) will help.



Now open the lowest menu item in LM's "**Settings**", the "**Expert settings**".

If you do not already have this item in your LM settings menu, you need to "long press" on the **Locus main menu button** and then tap on "**Expert settings**".

From now on, you will find the item by default in your LM settings.

In the "**Expert settings**" scroll down until you reach the section "**Hardware controller**".

There you make the following settings (*picture on the left:*)

For the **quick setting**, I have copied the **button** assigned with the function (*in my case*) and the **code** directly into the picture.

NOTE: Here you can of course assign the **buttons** on the RC to any of the seven functions in LM.

The *picture on the left* shows my key assignment

If you click on one of the sub-items, a window always appears (*picture on the right*) in which the **code** must be entered:

Here, in the example for the "**end side, lower**" **button**, enter **102** as the **code** and tap on **[APPLY]**.

The **code** of the **button** has now been assigned to this function in LM.

Proceed in this way for all six (operating) **buttons** on the RC.

You can now assign one of the seven different functions of LM to these six **buttons** (see *picture above*).

Finally, exit the "**Expert settings**" and the "**Settings**" menu via the app's "**Back**" button - no further saving or similar is necessary.



NOTE 1: in my experience you don't need to "configure" the joystick at all, it should scroll the map in the desired direction without any problems.

NOTE 2: There is a post in the LM forum that you can not only store one **code** for the "zoom functions" (In and OUT), as I have currently done, but that other numbers are also possible. Forum expert *freischneider* also reports on this. You can find the post in the forum here:

<https://help.locusmap.eu/topic/bluetooth-button-integration>

It could be that these "additional **codes/numbers**" control the size of the zoom steps or something similar. I don't know and, as already mentioned, I haven't entered them. In my opinion, the zoom works very well as it is...

NOTE 3: As already mentioned, you can of course also assign the **buttons** differently. The principle is now clear. It should also be obvious now how you can proceed/experiment with further "**presets**".

Step 4 - Restart Locus Map & Connect RC

Finally, close the "Locus Map" app and restart it.

Use of RC in Locus Map

The procedure for using RC in "Locus Map" will always be as follows in future:

1. Switch on RC
2. Connecting the RC to the smartphone via Bluetooth
3. Activate **Preset B** (*press & hold button M + press button B short*)
4. Launch Locus Map

Now the RC should work immediately 😊

uenu

03 .May 2023...*a humble forum member*