Samsung Gear360 Remote/Intervalometer

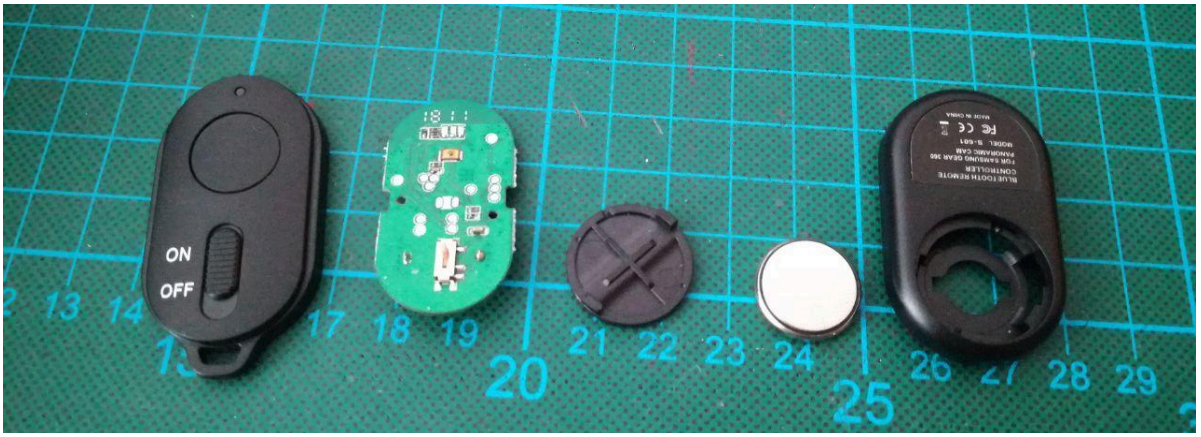
I've got a Gear360 2016 really cheap used from Amazon to play a little bit with it. The image quality is amazing for the price, but everybody who tried it knows that there isn't a good way to capture images in a regular interval.

Hardware Setup

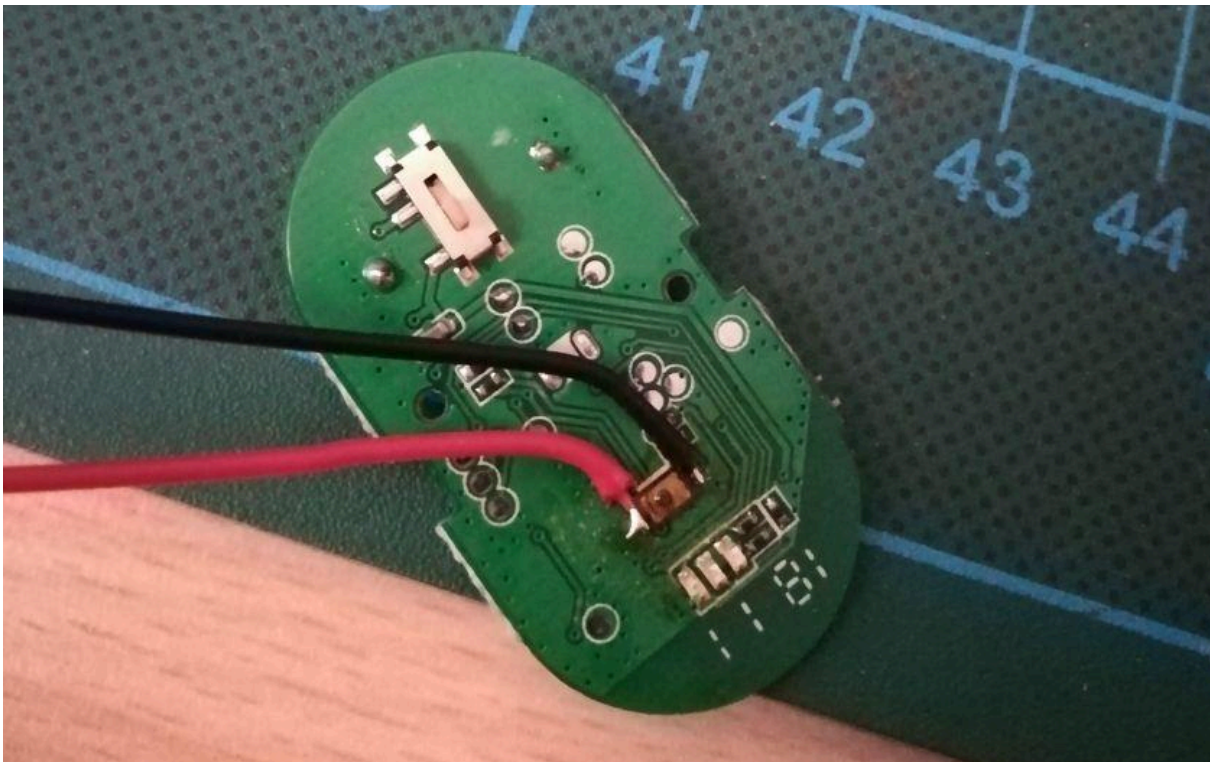
If you search for a few minutes than you stumble over this video https://www.youtube.com/watch?v=Py5VuD2KaAU where a remote shutter is soldered to the shutter button.
The idea to use a cheap intervalometer is great, but I didn't want to void the warranty and didn't like the cable on the camera. So I did the same modification but not to the shutter button on the camera but to the shutter button of a Bluetooth remote.
This is my remote.

I bought the cheapest Bluetooth remote on Amazon (something like this) and the cheapest intervalometer on eBay (something like this).

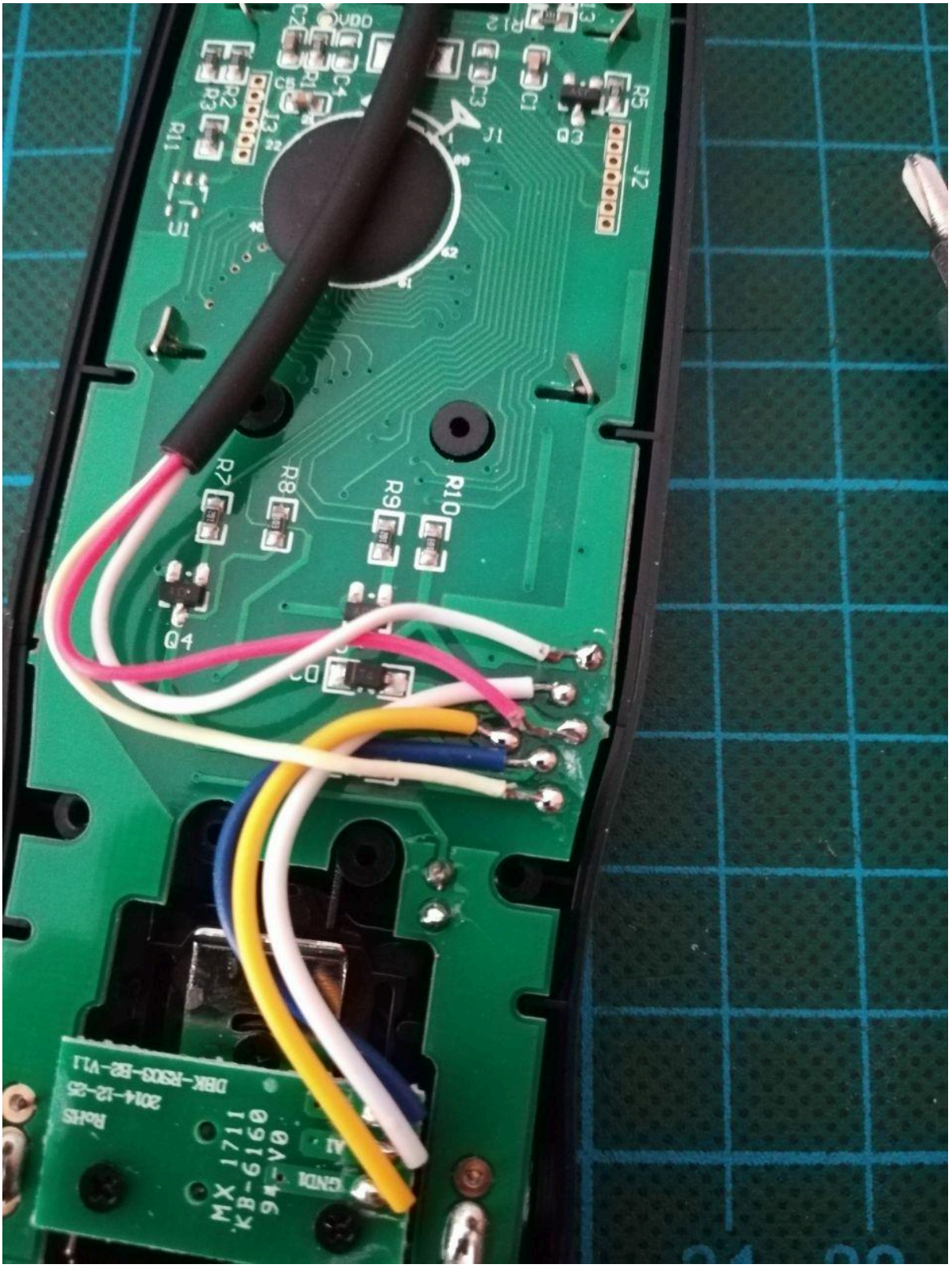
I disassembled the remote (you can pry it open with your finger nails or a knife)



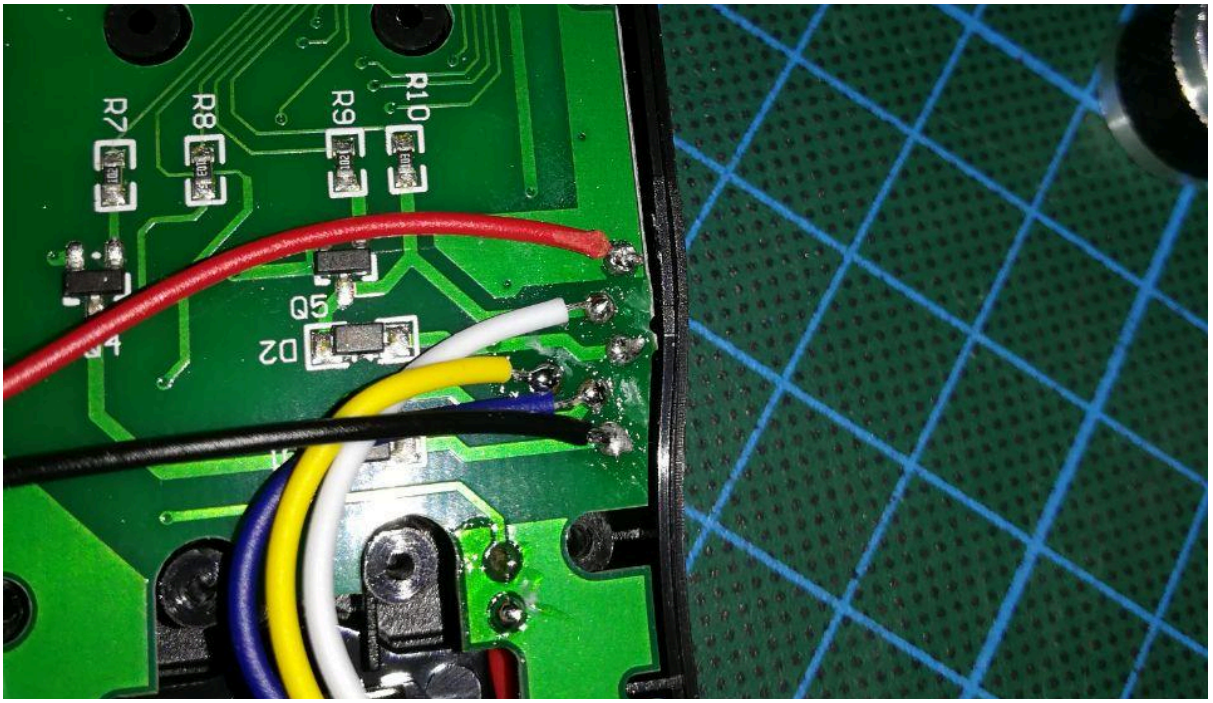
And soldered a wire to each side of the switch.



I opened the remote. (six screws: four on the back, two under the battery cover)



Now I desoldered the three wires from the big black external cable and just removed it completely. The two wires from the Bluetooth remote were soldered to two of the now empty pads like shown in the picture below.



To get the cases of both devices closed i cut small notche in each case for the wires and closed both cases up. As a last step I hot glued them together with a piece of plastic in the back for more strength (be carefully not to glue shut one of the two batteries compartments)

U can use the Bluetooth remote like before, but also with the intervalometer. Here is a short video to show it working.
https://www.youtube.com/watch?v=5Azzju7w30w

Software

Stitching of the images is a little bit tricky, the phone is too slow and I wanted a software solution which works on every platform. I found this project https://github.com/ultramango/gear360pano which uses hugin to stick the panoramas.
I used this script as base and added an automatic correction of pitch and roll with the help of the embedded meta data. I also added automated geotagging (I just have to find a good solution to compensate for difference between the gps track and photo timestamp).

I am not as good as a programmer to make a super clean program out of it. I will publish the scripts as soon as they are ready to everybody who is adventures enough to try my messy code. But I will give you a second way to try it. I've setup everything in a clean ubuntu virtual machine and I will upload it as soon as it is somewhat usable so you can just run the virtual machine without any more setup.

A first test sequence on mapillary will follow tomorrow or the next few days, my PC is stitching. I hope it turns out good.

Have a nice evening.