An antenna pointing system based on the u-blox C94-M8P-3 evaluation kit

Barry Chambers G8AGN

**Version 3, 1 Nov 2022** (additions to Conclusions section shown in red)

## Introduction

Over the years, when operating /P, I have tried a number of methods of pointing a microwave dish antenna in a required direction prior to making a contact. Correct alignment is particularly important when using electrically large dishes or when operating QRP, especially on mm-wave bands such as 122GHz. The most obvious aids to dish alignment are a good quality sighting compass and a sighting scope, fitted with adjustable cross-hairs, which is zeroed-in to the antenna boresight. Compasses work well if they have not been mis-treated and are used well away from large magnetised objects such as cars and farm gates etc. In addition, problems can be caused when operating over terrain which contains certain types of metallic ore. Using my high-quality **Silva**™ compass, I estimate that I can set a dish heading to within $2^{o}$ but often this is not good enough.

Alternative alignment techniques rely on being able to see approximately where the other station is and then using the sighting scope to position one's own dish accordingly. If the sky is overcast, then the distant station might make use of a lightbox consisting of a high-power red LED, such as one from the **Lumiled Luxeon**™ range, whose light is collimated using an inexpensive A4-size Fresnel lens (often sold as page magnifiers). If the day is sunny and the station to be worked is located North of your position, then a good and inexpensive alternative is to use an ordinary mirror to reflect the Sun's light in your direction. Light flashes from even a small mirror are easily visible to the naked eye over long distances. The problem of mirror aiming can be eased considerably by using a fire-ball aimer **[1].**

While looking for other dish alignment techniques recently, my interest was aroused by an article in DUBUS 3/2019 by David, VK3HZ, and Rex, VK7MO, in which they discussed the use of a RTK DGNSS (real-time kinematic, differential global navigation satellite system) setup based on a **u-blox**™ C94-M8P evaluation kit to align EME antennas when operating /P at 10GHz **[2]**. This technique seemed worthy of further investigation and resulted in this article.

## Background

I will not give a detailed discussion here of how RTK positioning systems work since there are several good YouTube videos available (for example, see **[3]**) but the point of interest to radio amateurs is that they can provide extremely high-quality information about the separation and compass bearing between two GNSS "modules", normally referred to as the Base and the Rover. This information may then be used to calibrate a rotatable protractor mounted on the operator's dish antenna tripod, such as that shown in **Figure 1**. Then it is a simple matter to point the dish accurately in the direction of a distant station whose beam heading has been determined previously using Google Earth or calculation from known Lat/Long coordinates or Maidenhead locators.


**Figure 1** Rotatable protractor on dish tripod (the protractor rollers and bed are made of PTFE)

Let us begin with the system geometry shown in **Figure 2**. Here the Rover is co-sited with the dish antenna and the Base is located some distance away, typically in the range 2 to 20m. Actually, it is the locations of the Base and Rover GNSS antennas which is important, since their phase centres are the reference points for timing or phase measurements. The electronics of the Base and Rover modules is identical and each contains a high-grade u-blox M8P GNSS receiver and a UHF radio transceiver (working in the 433MHz band for operations in Europe, 915MHz in the US). The radio link is used to send position correction data from the Base to the Rover. The distinction between a Base and a Rover lies solely in the way it has been configured in software, as will be discussed later.

In our application, the locations of the Base and Rover are interchanged from those of a normal RTK positioning system and in theory, at least, both the Base and Rover are not fixed and so the separation and angle between them can vary with time; this is known as the "moving baseline" mode. This is done for two reasons. Firstly, the Rover needs to be at the location of our dish antenna since this is where we will need to display to the operator the information about the angle between the Base and Rover; secondly, it is only when the system is configured to be in moving baseline mode that we have information from the Rover about where it is in relation to the Base. This information is expressed as two quantities dN and dE in Cartesian coordinates, as shown in **Figure 2**. (Actually, we have information available about this offset in three dimensions through a third quantity dD, for height, but this is not used).

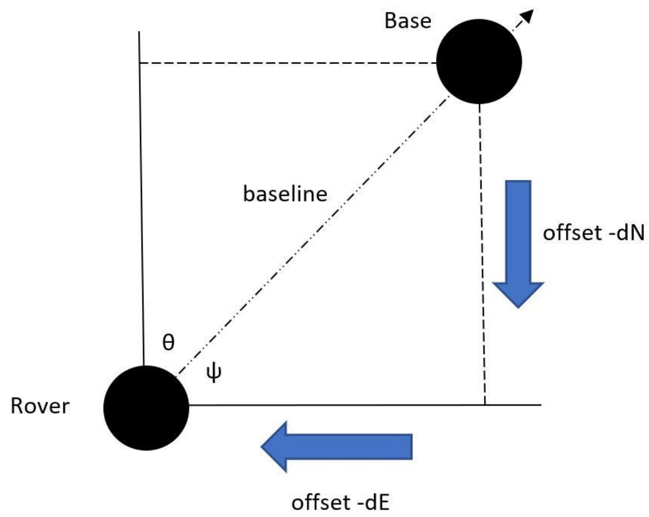


**Figure 2** Configuration of Base and Rover locations (blue arrows show displacements from the Base)

We need to calculate the angle θ, since this is the orientation of the Rover-Base baseline with respect to true North; it is given by

$$\theta = 90 - \psi = 90 - atan\left(\frac{-dN}{-dE}\right)$$

It should be noted that for this application, we do not need to know the location of the Base with very high accuracy since we are only interested in the relative location of the Rover with respect to that of the Base. Hence the Base location can be determined without recourse to a "Survey-In" procedure and an accuracy of about a metre will be adequate. The system should then be able to determine the length of the baseline to within a few mm and the baseline orientation to within about 0.1 deg. Once the latter has been determined, the dish antenna, with its associated sighting scope is then rotated to face the Base and the protractor rotated independently so its pointer indicates the value θ. The protractor is then locked into position on the tripod base; this completes the protractor calibration and now any dish pointing angle may be set accurately by simply rotating the dish so that the pointer indicates the desired angle.

Hardware requirements

**Figure 3** shows a C94-M8P module and its various interface and power connectors.

When used as a Base, there are only four connections to the outside world. Two of these are via SMA sockets into which are plugged the GNSS and UHF radio antennas; these are supplied as a part of the evaluation kit. The other two connections are for the power supply; the hardware can operate on any voltage between 3.7 and 20v. I chose to use a small 12v gel battery as the current required is modest, but a small LiPo battery would also be suitable.

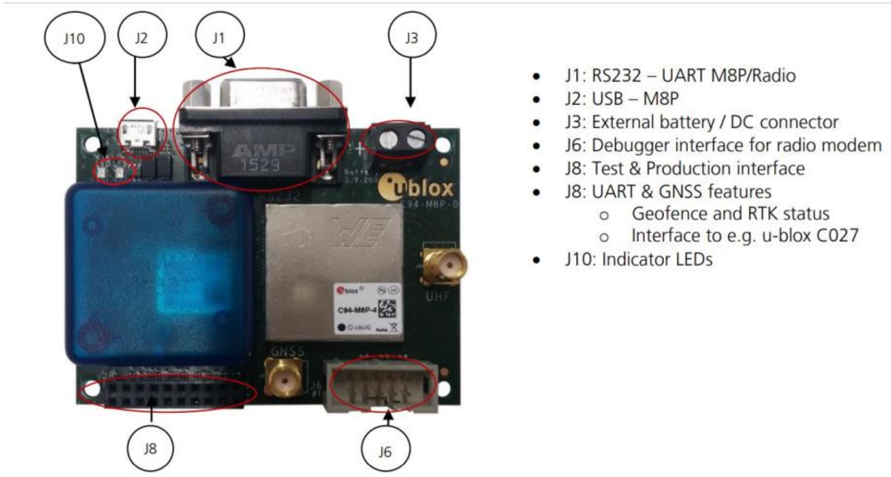


**Figure 3** The C94-M8P evaluation kit includes two of these modules. **[6]**

When used as a Rover, as well as the above connections, an additional one is required to carry data from the Rover, via J8 pin 10, to a serial interface input on an Arduino Nano or Uno as shown in Figure 4. This serial data is transmitted at 19200 bauds. The Arduino is used to parse the message data and to display the results on a 20 x 4 LCD display via a I2C interface.

It should be noted that the data passed from the Rover to the Arduino is not in the form of NEMA messages, but in a proprietary u-blox message format called UBX.

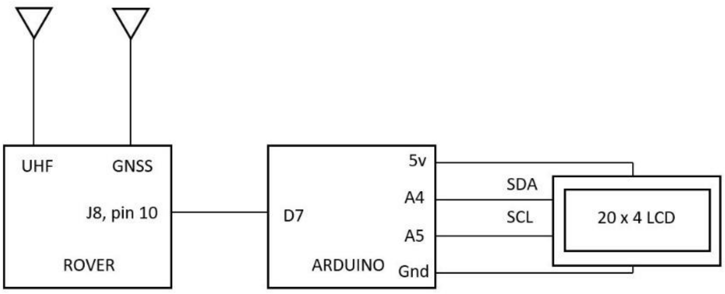


**Figure 4** Rover configuration (power connections to Rover u-blox module and Arduino not shown)

<u>Arduino sketch</u>

The Rover electronics, as originally described in **[1]**, used an Arduino Uno and a colour 320 x 240-pixel TFT display with integrated touchscreen. The latter was used to display two "screenfuls" of data, with the touch screen acting as a switch between the two. In my implementation, I opted to use a 20 x 4 I2C LCD display as it was cheaper, needed fewer connections to the Arduino, and was more readable in strong sunlight. I also replaced the Uno by a Nano.

The original second data display "screenful" showed a graph over a period of twenty minutes of the variation of the derived Rover to Base azimuth angle θ, thus providing a check on the (hopefully very small) azimuth angle variation with time. As my LCD display was not capable of displaying such a graph, this was replaced by a number representing the standard deviation of the azimuth angle variation over a period of one minute (based on a sample of twelve values of azimuth angle, collected at the rate of one every five seconds and stored in a circular data buffer). In addition, I also wanted my display to show the Maidenhead locator of the Rover; this facility was not available in the original system described in **[1]**. The resulting format of the information displayed on my LCD is similar in some ways to that shown on the first data screen described in **[1]**, but the absence of colour was overcome by displaying some information relating to the system status (for example, does the Rover have a valid GNSS fix) in either lower case (representing "no") or upper-case letters (representing "yes"). A typical LCD screen of data is shown in **Figure 5**.


**Figure 5** Rover LCD display

The top line of the display shows the Maidenhead locator and the standard deviation of one minute's values of the derived Rover to Base azimuth angle θ. The second line shows the current value of the azimuth angle in degrees. The third line shows the current derived value of the Rover to Base baseline length in m. The fourth line displays information about the Rover system status. FIX (in upper case) shows that the Rover has a valid GNSS fix. DIF (in upper case) shows that differential position corrections are being applied, REL (in upper case) shows that the relative position components (dN and dE) are valid and CAR (in upper case) shows that the GNSS receiver is operating using a carrier phase range solution with fixed ambiguities. The number (55) indicates that all the status flags described above are "true". When the Rover is first powered up, this number starts at 0 and increases in steps up to 55 as the various status flags become true.

A copy of my Arduino sketch can be obtained on request.

u-blox board configuration

As mentioned before, the two u-blox modules included in the C94-M8P kit are identical but need to be configured differently, so it is wise to label them as Base and Rover before proceeding further. A block diagram of the modules is shown in **Figure 6**, **[6]**.

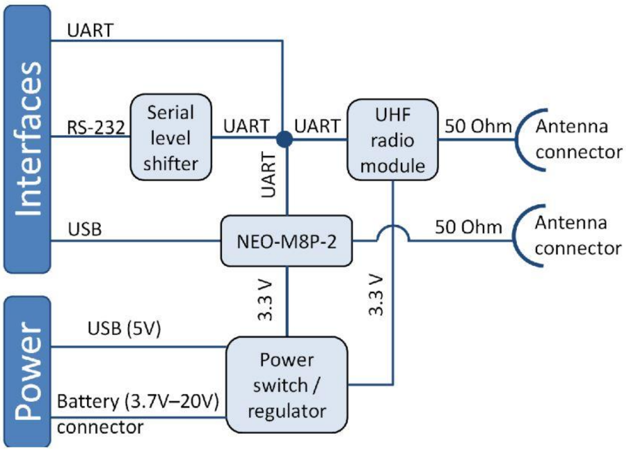


**Figure 6** Block diagram of the C94-M8P Base and Rover modules. **[6]**

Configuration is carried out by connecting a u-blox module to a laptop or PC via the USB interface J2, whose location is shown in **Figure 3**. The PC should be running the u-blox configuration software “u-center”, which can be downloaded from **[4]**. The latest software version, at the time of writing, is v22.05. Once a u-blox module is connected to u-center, it will act as a conventional GNSS receiver and transmit the usual NEMA messages; these can be viewed using the u-center “View” pull-down menu options. This data can also be viewed as lists/plots of visible satellites, position data etc. This step confirms that the u-blox module is working in its factory-default configuration. If, later, a mistake is made in the RTK configuration process, the module can be restored to the factory-default condition by pressing the “restore” button (cog-wheel logo situated near the top RHS of the u-center main screen).

Returning now to **Figure 6** for a moment, it can be seen that the actual u-blox M8P GNSS receiver is connected to the UHF radio transceiver and to the outside world via a UART. This will be referred to in the configuration documentation as UART1. This is also accessible via the multi-pin connector J8, shown in **Figure 3**. In particular, pin 10 of this connector provides us with a means of extracting 19200 baud-rate serial UBX messages from the Rover GNSS receiver which are subsequently processed by the Arduino. This facility is not used in a conventional RTK application and so will require additional configuration steps to be taken in the Rover procedure which is described below. These are not discussed in detail in either the u-blox documentation or in the DUBUS article **[1]**.

At this stage it is advisable to download the C94-M8P Setup Guide **[5]** and the C94-M8P User Guide **[6]**. The latter is a lengthy document but Section 4.3 (Moving Baseline RTK Configuration) is the relevant part. It may also be helpful before proceeding further to view a very helpful (but long) YouTube video **[7]** which shows the setup process for both the Base and Rover boards, as described pictorially in the quick setup guide **[5]**. As mentioned above, the steps shown in **[5]** do not quite correspond to those we require but this is clarified in the paragraphs below.

<u>Base configuration</u>

The Base should be configured as discussed in **[5]** but with the following modifications.

(1) In the u-center tool bar select “View”, then “Messages View”. Place your mouse over NEMA and right-click. Then click on “Disable Child Messages”. This will stop the GNSS receiver from outputting NEMA messages.
(2) Now go to Slide 7 in the quick setup guide **[5]**. This step is not needed so just check that the “Mode” box is set to “Disabled” (it should be, as this is the factory-set default).
(3) Turn now to Slide 9 and fill in the boxes as shown. Then press the “Send” button to send these commands to the GNSS receiver in the module.
(4) Fill in the boxes as shown in Slide 10 but replace the F5-05 RTCM3.3 1005 message with F5-05 RTCM3.3 4072.0 (the guide mentions messages as RTCM3.2 rather than 3.3 but this is OK). Note that each time you fill in the CFG_MSG window with new data in the boxes, you must press “Send” before moving onto the next data set.
(5) This completes the Base configuration procedure but before disconnecting the module from the PC, go to UBX-CFG-CFG and “save current configuration”.

The procedure outlined above is also explained in more detail in Section 4.3 in **[6]**.

<u>Rover configuration</u>

(1) Repeat step 1 of the Base configuration for the Rover module (turn off NEMA messages).
(2) Move the mouse down to UBX and right click. Then click on “Enable Child Messages”. This will command the GNSS receiver to only output u-blox proprietary UBX messages. A list of these may be seen by clicking on the + sign to the left of UBX.
(3) Turn to Slide 12 in the quick setup guide and fill in the boxes as shown, but for “Protocol out” select UBX. Press “Send”
(4) Select UBX-CFG-MSG. In the message dropdown list find “01-3C NAV-RELPOSNED”. Tick the box next to UART1 and change the 1 in the adjacent box to 5. This commands the Rover GNSS receiver to output a RELPOSNED message via UART1 (and pin 10 on J8) every 5 seconds. It is this information which the Arduino will process to provide the Rover-Base offsets and hence the baseline length and azimuth angle. Press “Send”.
(5) Select UBX-CFG-MSG. In the message dropdown box find “01-02 NAV-POSLLH”. Tick the box next to UART1 and change the 1 in the adjacent box to 10. This commands the Rover GNSS receiver to output a POSLLH message via UART1 (and pin 10 on J8) every 10 seconds. It is this information which the Arduino will process to provide the Rover Maidenhead locator. Press “Send”.
(6) Finally select UBX-CFG-CFG and “save current configuration”. It is now safe to power down the Rover module and the configuration process for both u-blox modules is now complete.

The interested reader will find detailed information about the UBX NAV-RELPOSNED and UBX NAV-POSLLH message formats in **[8]**.

<u>Using the C94-M8P system “in the field”</u>

To achieve optimum results, both GNSS antennas need to be mounted on ground planes. The ones supplied are just about adequate but ideally the Base antenna should be mounted on a larger one. Both antennas should have a clear view of the sky so as to access as many GPS and GLONASS satellites as possible.

Initial testing of the complete system was carried out “in the field” over nominal baseline lengths of 2m, 10m and 20m and with both GNSS antennas having a very clear view of the sky down to the horizon. The baseline lengths were measured out over fairly level ground using a 30m tape and the tripods supporting the Base and Rover GNSS antennas positioned accordingly; **Figure 7** shows the 10m baseline test configuration. The Rover electronics was housed in my car as shown in **Figure 8**. The Rover GNSS antenna is on the tripod nearest to the car and the baseline length could be changed by moving the Base GNSS antenna nearer or further away. The baseline orientation (Rover to Base) was pointing roughly East.

**Figure 7** 10m baseline configuration

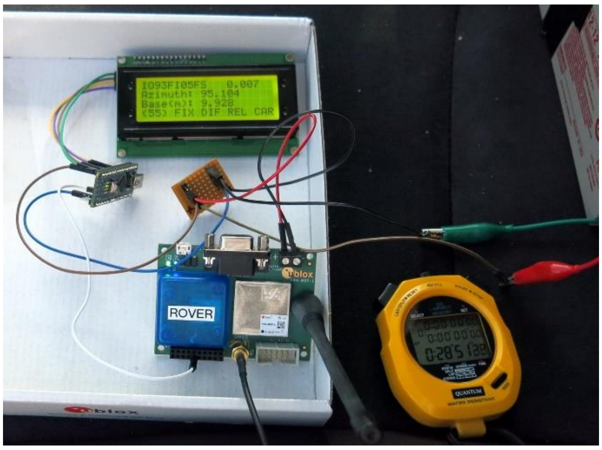


**Figure 8** Rover electronics package in the car (not yet properly boxed up!)

Each test was carried out over a 20-minute period after the system was first locked up and showing a status value of 55, indicating that a carrier phase range solution with fixed ambiguities had been obtained. Every minute the following data were recorded: time, status flag number (always 55), indicated azimuth angle θ in degrees, indicated baseline length in m and the one-minute standard deviation of the indicated azimuth angle values.

**Figure 9** shows the derived baseline azimuth angle variation with time for the three baseline lengths.

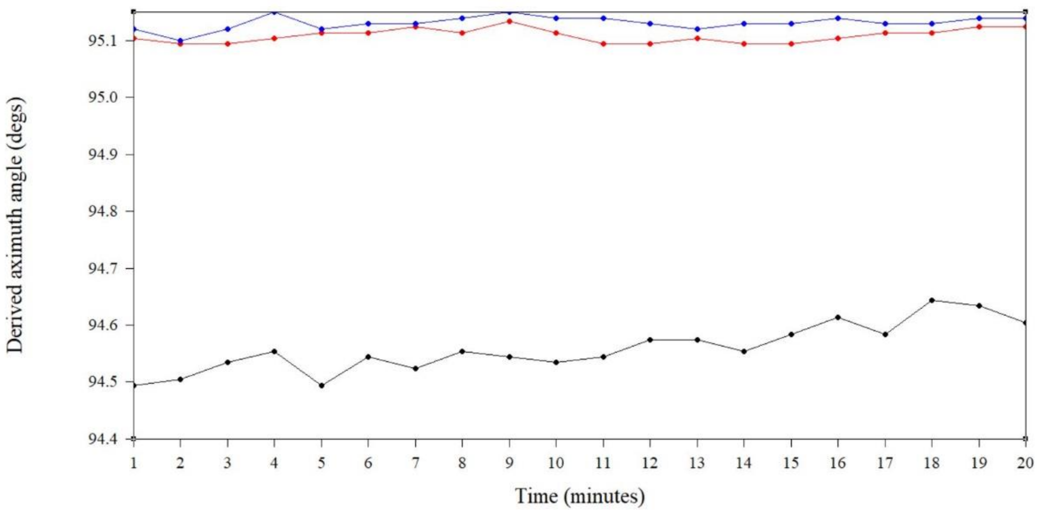


**Figure 9** Variation of derived baseline azimuth angle with time

Black curve = 2m, red curve = 10m, blue curve = 20m

The actual mean value of the derived baseline azimuth angle for the different baseline lengths is of no interest here since in moving the Base GNSS antenna tripod when altering the nominal baseline length between tests, some variation in tripod positioning transverse to the baseline direction was inevitable; this effect will become less important as the length of the baseline is increased.

What is of interest is the peak-peak variation of the azimuth angle with time. As might be expected, this becomes larger in the case of the shortest baseline, but even then, the variation is less than +/- 0.1 degree and for the 20m baseline the variation is approximately +/- 0.03 degree. Looking at the azimuth graph for the 2m baseline, the azimuth angle seems to increase slightly with time, unlike the 10 and 20m cases. It is suggested that this might be due to the tripod holding the Base antenna tilting slightly. The ground where the Base tripod was situated for the 2m test was uneven with long "tufty" grass. I did not "dig-in" the tripod feet and as the tripod had almost no weight on it, it is possible that over 20 minutes the grass "relaxed" and tilted the tripod very slightly in a direction transverse to the baseline. Only a small amount of tilt would be required to see the slight change in azimuth angle and the effect would be most evident for a 2m baseline. When positioning the Base tripod for the 10 and 20m tests, the ground was flatter with shorter grass and so the tripod was unlikely to move over time.

Although the data is not shown here, the derived baseline lengths (2.098, 9.938 and 19.890 metres, respectively) showed no variation over the 20-minute test periods. When considering the 1-minute azimuth standard deviation results, as might be expected, the largest values were found for the shortest baseline, but even then, the majority of the values were around 0.02 with the occasional value reaching 0.033, which is very acceptable.

A final comment to make about these results is that the complete set of tests took place over a period of almost two hours and so the GPS and GLONASS constellations as seen by the GNSS antennas would have changed considerably in that time; nevertheless, this appears to have had minimal effects on the results observed.

The next phase of the testing programme involved using a short rotatable baseline, as shown in **Figure 10**.


**Figure 10** Prototype of rotatable baseline

It was made from a length of timber 1.6m long, with a central pivot point and large steel washers were placed symmetrically at 25cm intervals. The latter were used to position the GNSS antennas and their associated circular ground planes. Initial tests were made with baseline lengths of 1.5m and 50cm. The test site was my back garden which is not ideal since it is only about 20m long and bounded by the house at one end (roof line maybe 10m) and trees at the other end (up to at least 15m?) Nevertheless, I was able to get a Rover 55 status after about 15 minutes and then it was good for about 95% of the test duration.

With this arrangement of rotating baseline, both the Base and Rover positions move as the timber spar is rotated. The new angle and baseline length were shown at the next display update but did take a few seconds to settle down. During and after a rotation the Rover status remained at 55 but as expected, the standard deviation value initially became very high (several 10s if the baseline angle of rotation was large ) before settling down over several minutes . The indicated baseline angle immediately after a rotation was within a degree or so of its final value and the indicated baseline was within 1cm of its nominal value. I tried to keep both GNSS antenna orientations the same with respect to the baseline axis so as to keep the antenna phase centre positions in roughly the same place.

By this stage in the testing programme, both Rover and Base modules had been boxed up properly, as shown in **Figures 11** and **12**.


**Figure 11** Rover module and LCD display

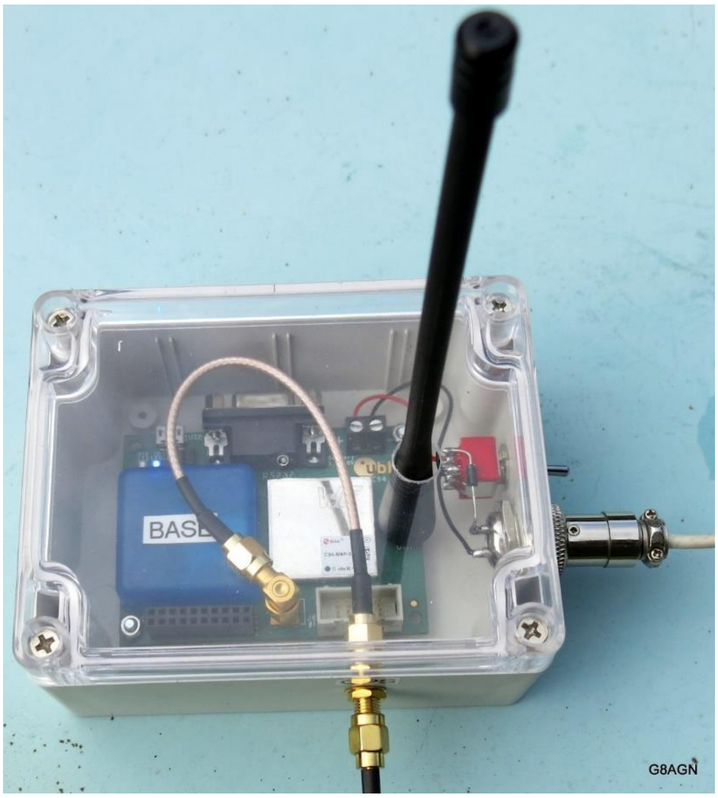


**Figure 12** Base module

The next test was made using the rotating baseline to see how accurate the GNSS indicated baseline heading was in comparison to optical sighting using a mobile phone base station antenna mast as the target. The latter  was approximately 10km away from the turntable. Google Earth was used to estimate the separation between the turntable baseline and the antenna mast and the latter's orientation. The baseline was then rotated to face the distant mast and aligned by sighting down the line joining the two GNSS antennas. This introduces a small angular error as it is not possible to know exactly where the GNSS antenna phase centres are within their plastic enclosures. Nevertheless, it was found that for a baseline length of 1.5m, the GNSS indicated baseline heading was within $0.5^{o}$ of that obtained using Google Earth. Using a baseline length of 0.5m, the error slightly larger but less than $1^{o}$.

In practice, the rotating baseline approach can be used in two ways. Firstly, the baseline can be established across the dish antenna aperture. Then the antenna heading is that of the baseline $\pm 90^{o}$. The choice of sign depends on whether the Base or the Rover GNSS antenna is located on the right-hand side of the microwave rig. This arrangement is shown in **Figure 13**.

**Figure 13** Rotating baseline parallel to dish antenna aperture

Alternatively, the baseline can be established along the dish antenna boresight direction, in which case the "GNSS pointer" gives the dish heading directly. One possible implementation of this case is shown in **Figure 14**, although others are possible. In **Figure 14**, the Base GNSS antenna is the one closest to the rig. If the rig were mounted behind the GNSS pointer, then the Base GNSS antenna would be the one furthest away from the rig.

**Figure 14** One possible geometry for a GNSS pointer

## Conclusions

From the work which has been carried out to date, the concept of a GNSS pointer seems to offer the potential for high accuracy dish antenna alignment when operating /P. The equipment is simple to set up and operate and can be situated close to large metallic objects such as cars. It does require, however, a fairly unobstructed view of the sky if a rapid RTK carrier phase range solution with fixed ambiguities is to be obtained.

Since the rotating baseline concept discussed above uses short baseline lengths, it might be possible to dispense with the radio link between the Base and Rover units and to send data via their hard-wired serial interfaces. Such a scheme could lead to the development of simpler and less expensive Base and Rover units but the development of these is left to others.

Finally, it should be possible to make the baseline arrangement shown in Figure 14 tiltable in elevation. Then use could be made of the dD component for the difference in height between the Rover and Base modules to calculate the dish elevation angle and to display it on the LCD.

Acknowledgement

I am grateful to David, VK3HZ, for his useful advice and suggestions concerning further testing while this project was being undertaken.

References

1. Making Your Mirrors - BSA Operation On Target (google.com)

2. "Differential GPS Azimuth Reference for Microwave Portable Operations", David Smith VK3HZ and Rex Moncur VK7MO, DUBUS 3/2019, pp 9 – 15.

3. How RTK works - YouTube

4. u-center | u-blox (u-center download)

5. Product Presentation (u-blox.com) (C94-M8P Application Board Setup Guide)

6. C94-M8P (u-blox.com) (C94-M8P User Guide)

7. Neo-M8P RTK setup and demo (courtesy u-blox) - YouTube

8. https://www.u-blox.com/docs/UBX-13003221