USING EXCEL MACROS FOR CHARTING

Bryan Kelly

Honeywell Aerospace

ABSTRACT

This paper introduces a set of macros that automate the importing of antenna data into Excel and charting that data. These macros (as discussed here) import data from a ViaSat ACUs (Antenna Control Unit) and a TCS ACU (Telemetry & Communications Systems Inc). After the import is complete, the macros can build a set of charts, all formatted and labeled in a predetermined and standard manner. A task that may take half a day or more can be completed in minutes. The concept and layout of the macros lend them to quick adaptation to your data. In scenarios of "test and collect" followed by "import and chart", the data can be imported and charted within the minute.

KEY WORDS

Date Reduction, Analysis, Charting, Excel

INTRODUCTION

The author is responsible for the performance of tracking antennas within a larger telemetry tracking system. At the end of a mission, logs of antenna performance must be analyzed and presented to show that the antennas tracked the target well. The file to be analyzed may contain fifty or more columns of data in a comma separated or fixed width file. The need for an automated utility was clear. This need for automated tools was fueled by the use of ACUs (Antenna Control Units) from two vendors, each with quite different logging formats for the same fundamental information.

Due to limited size constraints, these key topics are presented:

- ➢ The charts, the product of the macros
- ➢ How the charts are created
- ➢ How to control the charts

The reader will be able to obtain a user's guide, the macros, and example data from the ITC session information or from the author's personal web site. The user will be able to import the example date and create the example charts. .

Armed with this information, a moderately knowledgeable Excel macro programmer will be able to customize the macros to import other types of data and build charts as needed. An experienced Excel macro programmer can probably produce a customized version operable in a day. Less experienced users may require proportionally more time.

PRODUCT

The following illustrative chart was created from data provided by ViaSat. The data was collected by an elevation over azimuth antenna tracking a satellite. The X axis is time expressed in seconds beginning with the time the log file was started. That time was designated as time 0 (zero). The left Y axis provides the angle of the azimuth and elevation axes (in degrees) and the AGC level of the tracking receiver (in dB above noise). The right Y axis shows the tracking error of the azimuth and elevation axes in degrees. Note that monochrome printouts may be difficult to read.

ViaSat Sample Data Provided for ITC
22 - 25 October 2007
Macros developed by Bryan Kelly
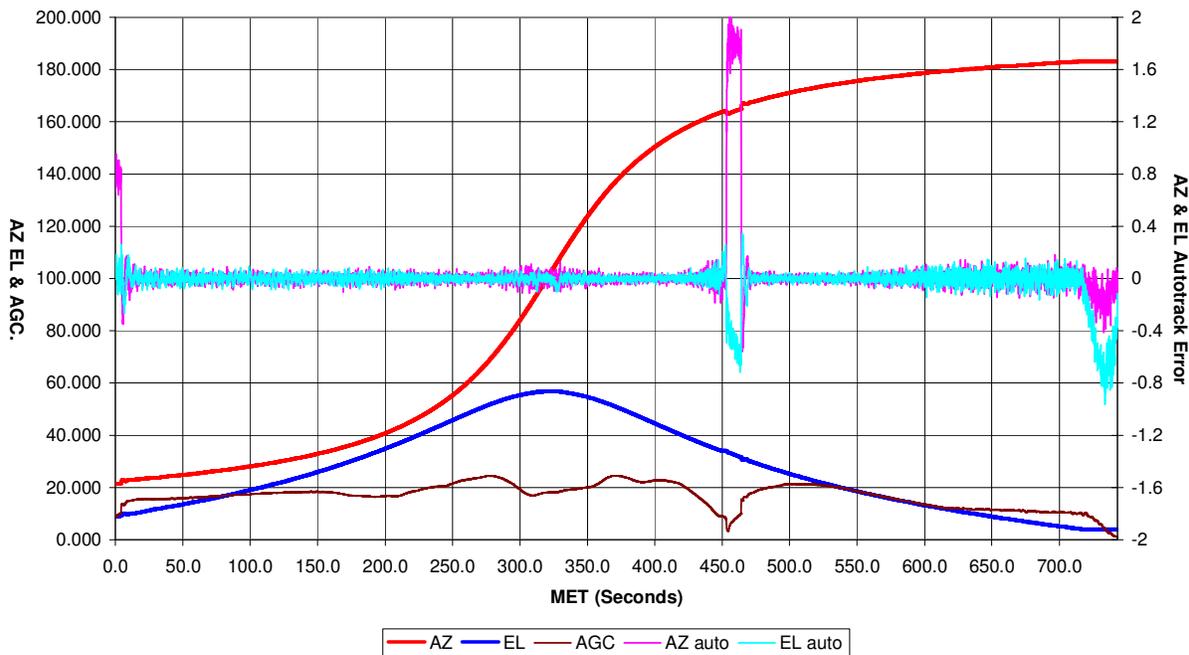www.bkelly.ws/macros macros@bkelly.ws

Figure 1 ViaSat Example Data

Tracking began at an azimuth close to north and swung around to south at the end of the pass. The elevation axis began at about 10 degrees, went up to almost 60, then back down to the horizon. The AGC or received signal strength moved up to about 18 at the start of the track and hovered around 20. For some reason the signal level dropped significantly at about 425 causing

the tracking error to increase. Tracking was recovered and continued through the end of the pass. We can see that for the majority of the satellite pass the tracking error was within 0.1 degrees. Once Excel is running and the raw data has been located, the data can be imported and the charts generated in less that than one minute. .

This example typifies how a satellite can be tracked and within a very few minutes, the data can be presented in charted format for easy analysis.

MACRO OPERATION

To produce the chart(s) the macros run through three program phases:
  - ➢ Import
  - ➢ Modification
  - ➢ Charting

The first two phases are generally performed as a single step, but can be separated for complex post import processing. The third phase produces standard Excel charts.

IMPORT

The import function begins by displaying a standard Windows open file control box. When the user has selected a file, the ViaSat and TCS macros import the data into Excel with a standard text or CSV import function. This import function has a multitude of options. A few examples include:
  - ➢ Rows to skip at the start
  - ➢ Data type (delimited, fixed width)
  - ➢ Delimiters
  - ➢ Type of data for each column (date, text, general)

The import operations that need to be performed depend on the characteristics of the data imported.

POST IMPORT

After the data has been imported, some processing may be needed. These macros were developed in the environment of tracking rocket and missile launches. In this field, a key synchronizing moment is T-0 or the point of first motion. For this type of data, the first item of post processing is to add an adjustable time field that can be synchronized with the vehicle first motion.

The data used in this paper is a 10Hz recording of dynamic antenna data such as azimuth, elevation, receiver AGC values and calculated tracking error. The data is column oriented with a new record, or row, being added ten times per second. Each record is stamped with the date and

time in a single field formatted as MM/DD/YYY HH:MM:SS.TTT. The time field contains seconds and thousands of seconds. Excel cannot deal with date and time in one field and it cannot manage the time in a format that contains colons and a decimal point. Excel under Windows 2000 will crash when attempting to access that type of data.

The macros import the date and time data into fields specifically declared as text fields to prevent Excel from analyzing them. The macros examine these time text fields, create a **Seconds of the Day** field and put this time in a new column with that title. A second new column named **Time Increment** is created containing time difference between each row. The final time column created is **Elapsed Time**. Working from the top, each cell of **Elapsed Time** is a simple Excel formula that adds in the value from that row's **Time Increment** to produce an incrementing time in tenths of seconds. A typical file will range from 0 to maybe 900 seconds in increments of 0.1 seconds. This becomes the X axis and is name MET or Mission Elapsed Time.

As the Elapsed Time field is a formula, the first row of data can be preset to a negative value. A carefully selected value will result in the Mission Elapsed Time counting from a T minus some value and having the Elapsed Time row containing 0.0 synchronized with the lift off or first motion time. The charts for our mission will then be synchronized with the controlling range's countdown clock.

The author wishes to note that while column names are changed for readability and consistency between charts, none of the original data is changed. If that data is not suitable for charting, a new column is created. This is a self imposed requirement to provide the original data in the worksheet without modification. The author suggests that all users continue and maintain this rule.

Some fields in the ACU logs are not formatted such that they can be charted by Excel. For example, some of the mode data columns contains data for two axes were in a single column. Other fields contained character representations of the ACU state. Excel cannot chart character based data. In all of these cases, post processing creates new numeric columns that can be charted and will convey the data intended in the original log.

Due to space limitations the different types of post processing cannot be discussed here. The point is that macros can be used to format and modify the data for better presentation.

CHARTING

The goal of these macros is to create charts that provide the user with the ability to extract information from large quantities of data. For each chart to be created there is a single macro that will find the data specified within the worksheet and create the chart. Once created, the colors of the chart will be specified, the labels of the chart will be created, and the chart worksheet will be given an appropriate name. One example chart was provided above. That example is shown as it was created by the macro. Once the parameters were set (as described below) the macro provided the final chart ready to evaluation or distribution.

This ability to create the charts as shown with no user modification is the primary purpose of these macros:

The process of using the macros presented two problems in giving the user control over the macros.

- ➢ The user must be able to select the appropriate macro to run
- ➢ The user must be able to control the display attributes

During development, the number of subroutines to create and manage the charts grew rather large. It became more difficult to find and run the appropriate macros. This was resolved by adding a menu item to the tool bar. Figure 2 below shows part of the Excel tool bar with the ViaSat menu installed. That menu item can be easily replaced by an appropriate menu for other applications.
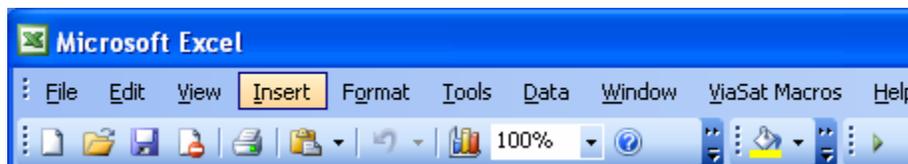

Figure 2 ViaSat Tool Bar

In the example provided here, the user has a new menu item named **ViaSat Macros**. Additional menu items can be added for different type of data. Clicking on this menu item opens a pull down with a two options. Figure 3 shows the ViaSat drop down exposed.

Figure 3 ViaSat Drop Down

The first task is to import the data. When **Import ViaSat Data File** is selected the macro performs the import function as previously described. The second selection of the new menu item is **Build ViaSat Charts**. When that option is selected a form is displayed as illustrated below in Figure 4
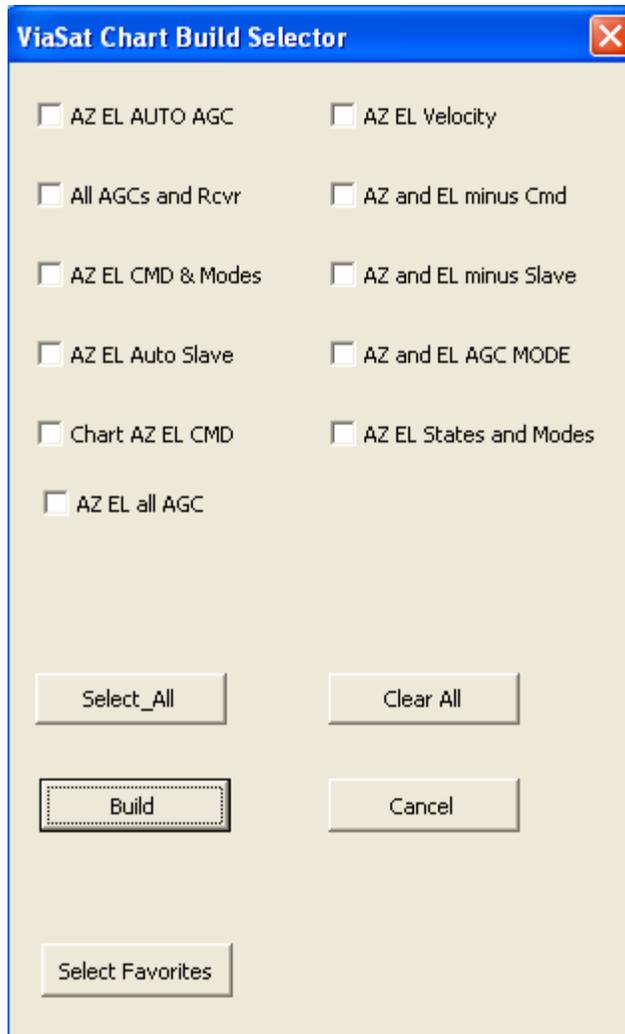
Figure 4 Chart Selector Form

This menu provides a check button for each chart that can be created. The user selects the ones desired then clicks on **Build** to construct the selected charts. The Select All, Clear All, and Cancel buttons perform the obvious functions. The macro programmer can pre-select a set of most often used charts and the **Select Favorites** button will select those charts for building.

FORMATTING

The second problem mentioned earlier is control over the formatting and labeling of the charts. Each set of data may present its own unique ranges of the various data items. The solution is to create a worksheet that controls how the charts are scaled and colored. The import function creates a worksheet named **notes**. The reader is advised to open one of the example data sets and examine the notes sheet for this discussion.

TITLE

There are three groupings within the **notes** worksheet. The first is title and date and occupies the first six rows within the worksheet. Those rows are illustrated here.

This sheet must be named notes
These cells are used by macros to set chart parameters
Set the T-0 in the Elapsed Time column before proceeding
Title
Demonstration Data for ITC
22 November 2007

The first four rows are reminders to the user. The next two lines become the title of all charts.

SCALING

Immediately following the title and date in the **notes** worksheet is the "scaling" information. The first of these must be the horizontal scaling and is illustrated here:

Notes_X_Label
Minimum   0
Maximum   1097
Major Units   50
Crosses   -40

To simplify the discussion, the above five lines of text will be referred to as a **scaling declaration**, or more simply, a **declaration**. The first line of each declaration is its name. This name is also the name of the constant that the macros use to find this data. The end user will not be concerned about this constant, but the macro programmer will find it extremely useful. When specifying a data group to use for scaling a given data series, the programmer will simply copy the text in this first line into the code as the name of that group.

Returning to the horizontal declaration, the **Minimum** and **Maximum** values here become the limits of the X axis. In the TCS and ViaSat macros, the import function examines the data and determines the range of the entire data set. Those range limits are put in the horizontal declaration during the import phase. The first charts will show all the data in the worksheet. After the import phase is complete, the user may edit these values as needed.

The **Major Units** value controls the interval markings along the bottom of the chart. The **Crosses** value controls where the Y axis intersects with the X. These are standard Excel values and are not elaborated here.

Following the X axis declaration is an unspecified number of Y axis declarations. The macro programmer may change the number and names of these declarations and may edit the initial values as desired. A typical Y declaration is provided here.

Notes_AGC_Values_Label

Minimum   10
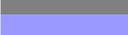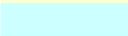Maximum   60
Major Units   10
Crosses   0

The ACU macros use one declaration for each type of data found in the log file. Note that the format and significance of each value is the same as for the horizontal (or X) axis declaration but transposed to the Y axis. In the ViaSat macros, at this writing there were eleven declarations. Note again that the first row of the group contains the name of that **declaration.** This name is also the name of the constant used in the macros reference this declaration.

The Excel macro programmer will edit the macros to control the fields written to the **notes** worksheet. The programmer has the task of configuring the notes worksheet according to the needs of the user. The user will edit the values in the notes sheet to make the final determination of colors and ranges for their charts.

LINE COLORS

The third part of the notes sheet controls the colors of each charted data item. The header of this section displays the colors that are available to the user. Table 1 is an excerpt from this section of the **notes** sheet.

Table 1 Color Reference

1	11
2	12
3	13
4	14
5	15
6	16
7	17
8	18
9	19
10	20

For each color that is available to the user, one cell of the **notes** sheet is set to that color and the number of that color is provided to its left as a reference. This section is for user reference only and is not utilized by the macros. Its purpose is to relieve the user of the task of remembering the number that Excel has assigned to each color.

The next six lines typify the entries of the color control section.

Name	Color	Width	Line Width Values
Background	2		1 = Hairline
AZ_Color_Row	3	3	2 = Thin
EL_Color_Row	5	3	3 = Medium

Axis3_Color_Row	4	3	4 = Thick
AZ_Cmd_Color_Row	4	2	

In this example there are four column headings, Name, Color, Width, and a fourth column for informational purposes only. The background color is always present and in this case is set to white (Reference the color chart and note that Excel uses the value 2 for the color white). All charts are built with this specified background color. Following that is a series of one row entries, one for each type of data item that is charted. As before, the first column contains the name of the constant that selects that row. The second field is the number of the color that will be assigned to that data item. The third field controls the width of the line. A small sub chart under the heading **Line Width Values** is contained within notes to remind the user of the available values for the line widths. Macro writers are strongly advised to have one entry for each type of data item that can be charted.

As discussed here, every time azimuth is charted, it will be assigned the color 3, red. Elevation will always be blue. This standardization of colors will make it easy for the chart readers to move from one chart to the next and be able to make comparisons between charts.

CONCLUSION

This paper has described a utility and a method that can be used to automate the charting of large amounts of data. The three major impediments to automated charting presented here were: column specific macros, building the chart, and controlling the chart format. The basic concepts have been discussed, but due to space limitations, the reference material must be consulted for details. A moderately experienced Excel macro programmer can customize these macros for individual and company use.

ACKNOWLEDGEMENTS

ViaSat and TCS have provided example log files expressly for this ITC paper. I sincerely appreciate their assistance. The ViaSat URL is www.viasat.com. The TCS URL is www.tcs.la.

My most important asset in creating these macros is the extraordinarily knowledgeable and patient members of the forum www.MrExcel.com.   I could not have created these macros without their extensive help. To all the members of MrExcel.com, I extend my sincere appreciation. Readers are encouraged to search and participate in this forum for Excel questions.

REFERENCES

The following is available from the author's web site at www.bkelly.ws/macros.

Macro User's Guide: a user's guide written for these charting macros.
The macros in source code form: the source code of the macros for the TCS and ViaSat ACU.
The macros in an Excel workbook form.
Example data: example log files provided for this paper by ViaSat and TCS.

Example workbooks: Two workbooks, one each containing the example data from ViaSat and TCS. A few charts are provided in each workbook.

The author may be contacted at: **macros (at sign) bkelly dot ws**. Two symbols have been replaced with their names and some spaces to prevent web bots from extracting and spamming this address.

NOMENCLATURE

ACU   Antenna Control Unit
AGC   Automatic Gain Control
ViaSat   A satellite and communications company
TCS   Telemetry Communications Systems Inc.