

μTracer 3.10 User Manual

Contents:

1. Introduction
2. Getting started
 1. The GUI in overview
 2. Switching on the heater
 3. Measuring a set of output curves
 4. A quick test
 5. Saving the pinning and measurement setup
3. Measurement Set-Up
 1. Measurement set- up overview
 2. Measurement types
 3. Averaging and Ranging
 4. Compliance
 5. Delay
4. Graphical Output
 1. Controlling the Graphical Output
 2. Transconductance and plate resistance
 3. Keep Plot
 4. Matching of tubes (storing a graph)
 5. Transconductance and plate-resistance
5. Quick Testing
 1. Quick Test introduction
 2. Using Quick Test
6. Saving Plots and Measurement data
 1. Saving a graph
 2. Saving measurement data
 3. Importing data in Microsoft Excel
 4. Saving the measurement setup
 5. The calibration file
 6. Location of the files
7. Installation of the GUI
 1. Trouble shooting
 2. Installing a new release
 3. Testing the USB to Serial converter
 4. Stacking of COM ports
8. Capita Selecta
 1. The communications form
 2. Magic eyes & continuous measurements
 3. Heater considerations
 4. The calibration procedure
 5. Positive grid bias (A2 mode)
 6. Suppressing oscillations



1 - Introduction

To keep the hardware of the uTracer as simple as possible, the complete operation of the uTracer is performed under software control. The program which controls the uTracer is called the Graphical User Interface (GUI). The GUI is a standalone program that can be downloaded for free. The GUI communicates with the uTracer through a serial data link.

The GUI was in the first place designed to trace the curves of tubes. These curves give all the information about the condition of the tube and provide information for the design of circuits. In addition to the curve trace function, a tube testing function has been implemented which measures the most important tube parameters for a single bias point. This “Quick Test” feature has been implemented in GUI versions 3p10 and higher.

The operation of the uTracer is pretty straightforward and usually consists of four steps: 1. setting up the measurement conditions, 2. perform the actual measurement, 3. Adjusting the graphical representation if needed, and 4. storing the data. Once the GUI has been set-up for a particular tube, the complete measurement set-up can be saved to a file so that it can be recalled at a later time. The Quick-Test is even simpler: just enter your favorite bias point and nominal anode current and press “Test.”

The operation of the GUI is rather intuitive. The “getting started” section of this manual will get you going as quickly as possible without going too much into details. There are however a few controls and features that need some more explaining. They will be dealt with in subsequent sections.

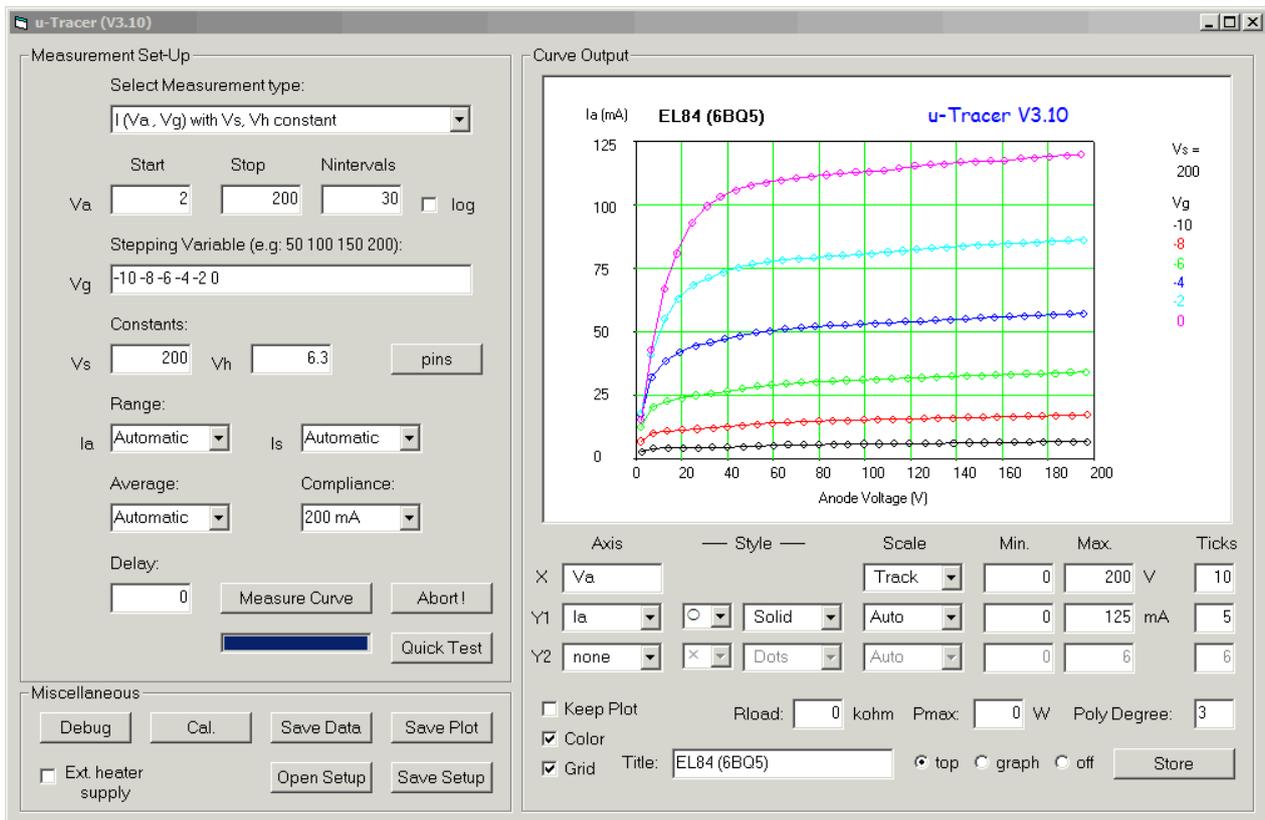


Figure 2.1 Main form of the uTracer 3.10 GUI.

2 - Getting Started

In this section it is assumed that the GUI (version 3p10) has been installed and that the link with the uTracer hardware is functioning. To get some hands-on experience it is a good idea to have a medium-sized pentode like an EL84 (6BQ5) connected to the uTracer. You can also use a triode, but that case it is not possible to explore the screen current measurement options.

2.1 - The GUI in overview

Figure 2.1 shows the main “form” (a Visual Basic term) of the GUI. There are three areas (frames) on this form. The upper-left area is for the measurement set-up. Things like the type of measurement, the voltage ranges and details about range selection, averaging and current compliance can be set here.

The right side of the form is for the graphical display of the measured data. Here is specified what is plotted, and how it is plotted. By default the anode current is plotted along the left y-axis. It is also possible to select another parameter, or to activate a second y-axis on the right e.g. to simultaneously plot the screen current. Furthermore, the scaling of the axes, marker type, line style, plot title etc. can be set. By default the scaling of the axes is set to automatic so that for a quick plot you don't have to set anything.

The small section at the bottom-left is reserved for all the other stuff such as storing the measurement data or plot, storing and retrieving a measurement set-up, modifying the calibration values and debugging communications.

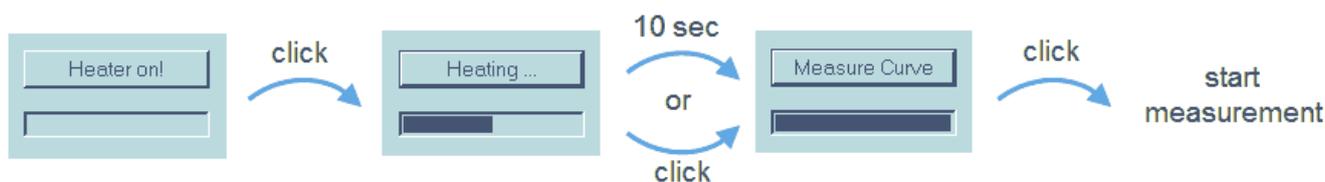


Figure 2.2 Switching on the heater.

2.2 - Switching on the heater

Since it takes some time for the heater to switch on and stabilize, it is a good idea to first switch on the heater, and then adjust the other measurement setting. Before you switch on the heater make sure that the proper heater voltage is displayed in the field marked “Vh”. If it is not, adjust the value to any value between 0 and the supply voltage of the uTracer, usually something like 19.5V. When you are satisfied that the correct heater voltage is set, press the button marked “Heater on!”. The text on the button will now change to “Heating ...”. The GUI will start sending commands to the uTracer hardware to gradually increase the heater voltage from zero to 100% of the set point value. This “slow-start” reduces the thermal shock normally associated with switching on a cold heater from a low impedance voltage source. The progress-bar underneath the command button indicates the progress of this process.



Tip: pressing the command button again while it is displaying “Heating ...” will directly switch the heater voltage to the specified set-point value.

When the heater is switched fully on, the text on the command button changes to “Measure Curve”. In principle a measurement can now be started, however, it can take up to a minute or so before the heater reaches its nominal operating temperature (and thus emission). So we will use that time to set-up the measurement.

2.3 - Measuring a set of output curves

While the heater is stabilizing, the rest of the measurement set-up can be finished. The uTracer is capable of measuring and drawing any of a number of different types of curves, which can be selected from the drop-down menu at the top of the measurement section of the main form. To measure a representative set of output curves, select the third one which is marked: **$I(V_a, V_g)$ Vs, V_h constant** (Fig. 2.1). This is short hand for: ‘measure the currents as a function of the anode voltage (x-axis), stepping the grid voltage, while keeping the screen and heater voltages constant.

When a new measurement type is selected, the GUI will automatically set the bias conditions to default values, which are usually a reasonable starting point. For a set of output curves the default screen voltage is 200 V, while the anode voltage is swept between 2 and 200 V. Any value between 2 and 300 V can be entered in these fields. In this example we may want to add some grid bias values, so replace the -10 V for the following series separated by spaces: “-10 -8 -6 -4 -2 0” (Fig. 2.1). After these preparations start the measurement by pressing the commandbutton which now displays “Measure Curve.”

The high-voltage LED will come on, indicating that high voltages are present in the uTracer circuit, and the uTracer will start acquiring data for the output curves. As the curves are being measured, they are plotted in the graphical output section of the main form. Normally there will be little need to change the settings in the output section since the axis will be automatically scaled as the measurement progresses. In [section 3](#) the purpose of the different controls is explained. The measurement can be interrupted by pressing the “Abort” button. After the measurement (or after an abort), the uTracer hardware will discharge the high-voltage electrolytic capacitors to a safe level and the high-voltage LED will turn off.

Tip: do not initiate a new measurement before the high-voltage LED has turned off because this may result in a “hang-up” of the software which will require a restart of the hard- and software.

If you have connected a pentode (or tetrode) select parameter “Is” from the drop-down menu in the column “Axis” next to “Y2”. The screen current will now be added to the display, along with the axis on the right-hand side of the graph. Clicking with the mouse on or near one of the curves will cause a marker to appear. The value at that point will also be displayed, interpolated between measurement points if necessary. To save the graph, click on “Save Plot” in the “miscellaneous” section of the main form. A new form will open prompting you to enter a file name and a path. Unfortunately the only format supported is “.bmp”. The measurement data can also be saved in ASCII format for further processing by pressing the “Save Data” command button. A variety of data formats are available which are discussed in [section 6.1](#).

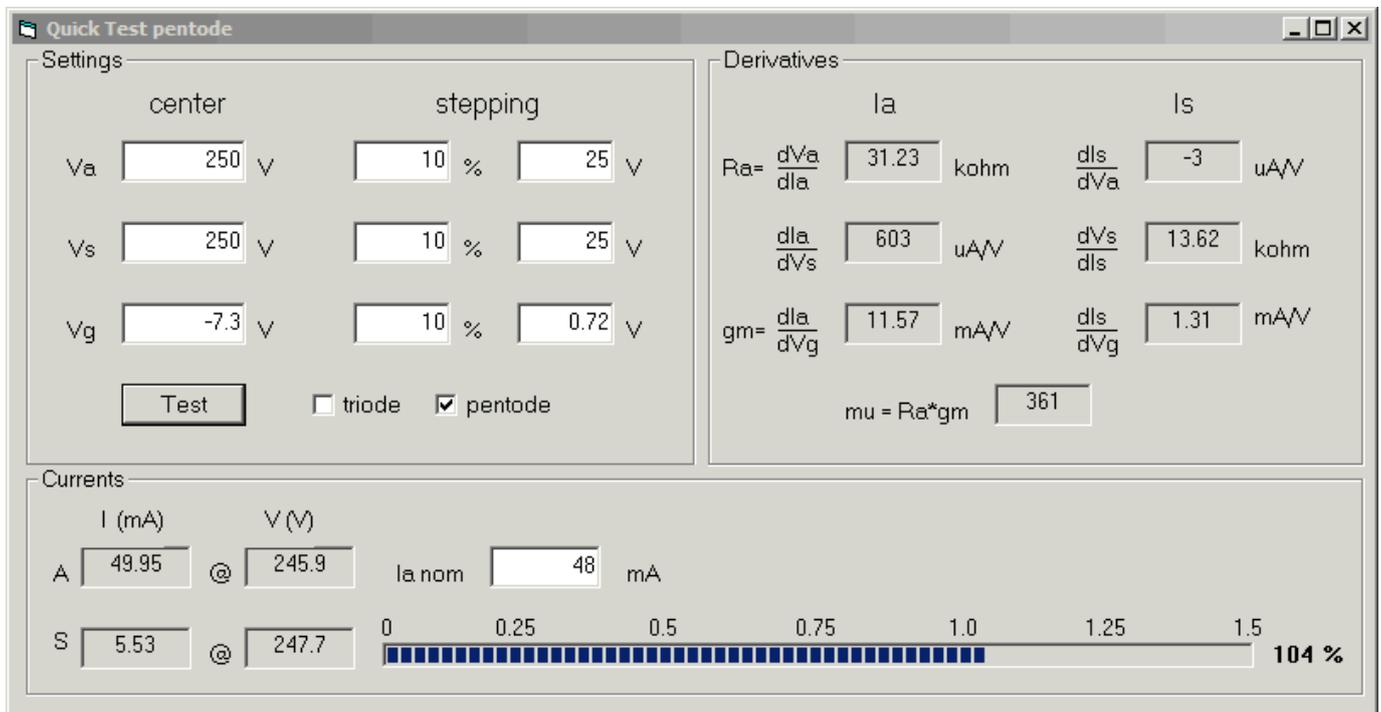


Figure 2.3 The Quick Test.

2.4 – A quick test

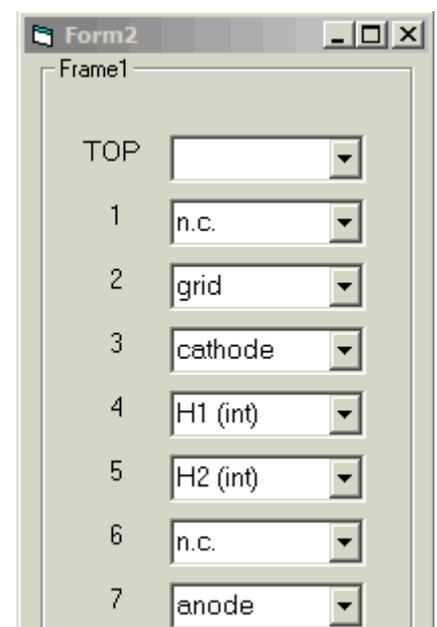
Apart from taking full curves, the uTracer can be used to measure the most important parameters such as plate resistance (R_p), transconductance (G_m), amplification (μ) and currents at a single bias point. Manufacturers of tubes often give these parameters at a bias point which they consider to be optimal. Since only a few measurements are needed to extract these parameters, a “Quick Test” only takes a few seconds.

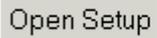
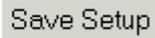
With the heater on, the quick test can be invoked by clicking the command button labeled “Quick Test”. A new form will be displayed from which the quick test is performed. If you have hooked up a pentode to the uTracer, tick the “pentode” box, and an alternate form will appear. Now enter the bias point for the quick test. In case of an EL84 this would typically be $V_a = 250$ V, $V_s = 250$ V, and $V_g = -7.3$ V. Under these bias conditions the anode current according to the manufacturer should be 48 mA, so this value is entered in the box labeled “ I_a nom.” Now click “Test”. The GUI will now measure the currents in a few bias points in and around the specified point and extract the derivatives.

More details about the quick test function will be discussed in [section 5](#) of this manual. For now I would only like to add that if double-triode is to be tested, both sections can be tested simultaneously. In this way information about the matching of the sections is quickly revealed.

2.5 – Saving the pinning and measurement setup.

Depending on your personal preferences you will have chosen a means to connect the uTracer to the pins of the tube sockets. Some people use a simple “banana plug” type of connector while others prefer rotary switches or a similar solution. Either way, it is useful to have a means to store information specifying which pins the terminals of the uTracer are to be connected to. GUI versions 3p9 and higher have a form on which this information can be stored. The form can be invoked by clicking the “pins” button. It displays the pin numbers, and next to each one a drop-down menu from which the appropriate uTracer terminal can be selected.



A rectangular button with a thin border and a light gray background, containing the text "Open Setup".A rectangular button with a thin border and a light gray background, containing the text "Save Setup".

Note that a number of extra and spare terminals have been included for future expansion. *This feature has only been added for the convenience of the user and in no way effects the working of the uTracer or the measurement.*

At this point it is possible to save the entire measurement setup to a file so that it can be retrieved later, i.e. when another tube of the same type is to be measured. To save the set-up, simply click “Save Setup”, and enter an appropriate file name, e.g. “EL84out”. The file containing the set up data has the extension .uts, and it will be stored in the same directory from which the GUI was started unless another path is specified. The set-up file is a normal ASCII file and may be examined using any text-editor.

3 – Measurement Set-Up.

This section discusses the “Measurement Set-Up” form in detail. The main features of the form will be presented in an “Overview” section, and in the following sections features needing a more in-depth explanation will be reviewed.

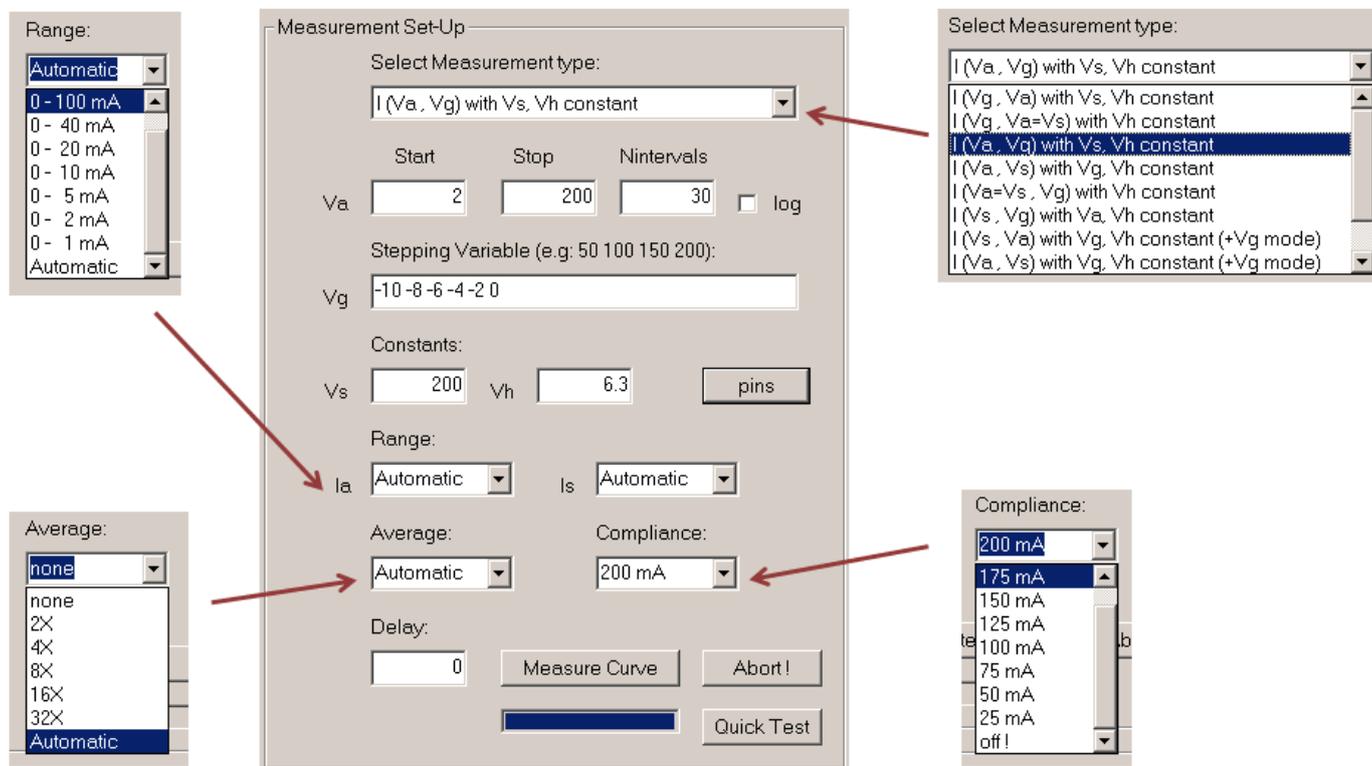


Figure 3.1 The “Measurement Set-Up” form

3.1 – Measurement set-up overview

Setting up a new measurement is quite simple and intuitive. The best approach is to go through the measurement set-up form from top to bottom. Most settings have been pre-programmed or are automatically filled-in when a new measurement type is selected, so that usually very little adjustments are needed to produce a set of curves.

The first thing to specify is the measurement type. This is done by selecting one of the measurement types from the drop-down menu on the top of the form (Fig. 3.1). In GUI 3p10 ten different measurement types have been predefined which will be reviewed in the next section. Note that when a new measurement type is selected, the variable names and the default values in some of the other text boxes will change appropriately.

Every measurement involves a “running” variable and a “stepping” variable, and possibly one or two constant voltages. The running variable is the variable along the x-axis. This variable is specified as a range from “Start” to “Stop” comprising a number of measurement points “Nintervals.” Normally the distance between these points is equidistant, however, by ticking the box marked “log,” a so called logarithmic sweep is generated whereby the distance between the points increases along the curve (Fig. 3.2). This feature is used to generate more points in the beginning of the curves where the gradients are usually highest.

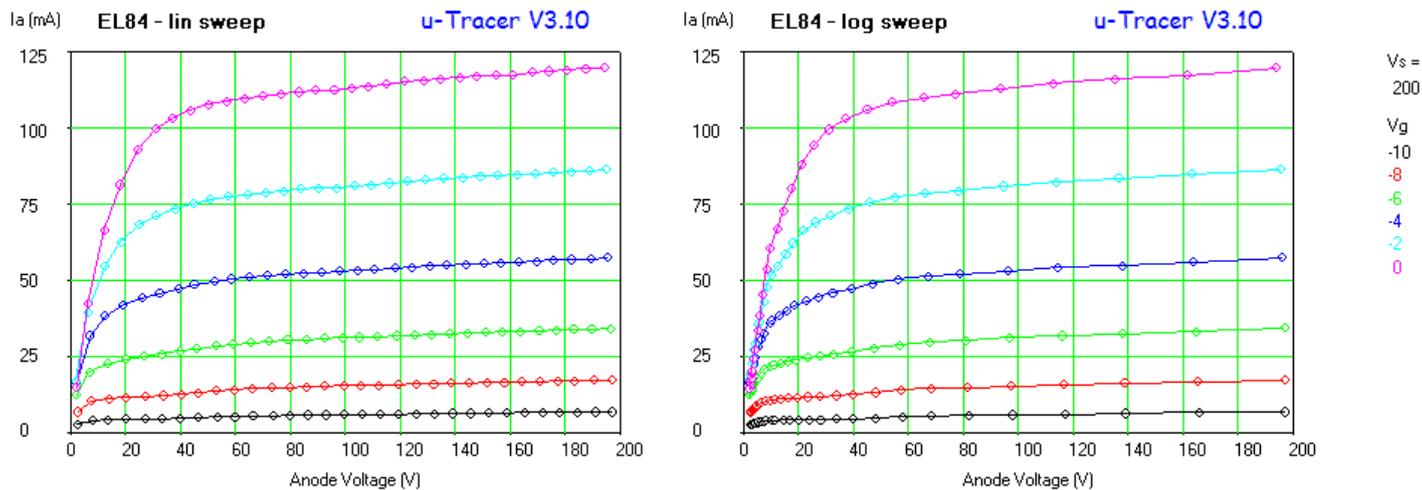


Figure 3.2 The difference between a linear and a logarithmic sweep

In the text field below the running variable the stepping variable is specified. Up to twenty values can be entered, separated by spaces. The stepping variable values are displayed to the right of the plot as the measurement progresses. Below the stepping variable there are usually one or two text boxes for constants.

The following ranges apply:

- High voltages (anode and screen): 2 to 300 V
- Control grid: 0 to -50 V (the minus sign must be included)
- Heater voltage: 0 to V_{supl} (the uTracer supply voltage)

A lot has been said and written about the heater voltage. It appears that for very low heater voltages and high currents the actual heater voltage can be lower than the voltage specified. In a [separate section](#) this issue and some possible workarounds will be discussed.

In most cases after setting these values a measurement can be started by switching on the heater and starting a measurement as described in the [“getting started” section](#).

Good to know:

Under some conditions the high voltages actually applied to the tube can be slightly less than the voltages entered in the GUI! The reason is related to the way the uTracer works: before each measurement, the reservoir capacitors are charged to the specified set-point value. When the set-point value is reached the high voltage switch closes and the tube is connected to the reservoir capacitor. After a stabilization time of 1 ms the anode and the screen current are measured. Unfortunately the actual voltage at the terminals is slightly less than the original set-point voltages. This is because the high voltage switch and the current sense resistor both introduce small voltage drops, and also because the reservoir capacitor is slightly discharged during the 1 ms stabilization time! The actual terminal voltages can therefore be several volts less than the set-point values. For this reason the *actual* voltages at the end of the measurement pulse are measured. It is this measured value, corrected for the voltage drop over the switch and the current sense resistor which is used as the x-axis variable. So, suppose the set-point value was 250 V and the measured value was 247 V then the last value is used for the plot. On the “Debug” form a box can be found labeled “Voltage Corr.”. By default this box is ticked, indicating that the voltage correction is on by default. If so desired it can be switched off.

So far, so good. However, when the high voltage in question is used as a stepping variable, or as a constant, there can be a small difference between the set point value and the actual voltage applied to the tube. In the “Quick Test” forms by the way the “real” voltages are

displayed after the measurement. This makes it possible to correct for the voltage drop by selecting a slightly higher set-point value. One of the improvements for a next GUI version might be a provision to correct for this small error.

3.2 – Measurement types

The type of measurement can be selected in the top drop-down box of the measurement section. In GUI version 3p10 ten measurement types have been defined. A kind of shorthand notation is used to identify the measurement types. For example:

I(Vg, Va) with Vs, Vh Constant

means: measure the currents – the anode and screen current are always measured simultaneously – with Vg as running variable (along the x-axis) and stepping Va, while Vs and Vh are being held constant.

The following measurement types have been defined:

1. **I(Vg, Va) with Vs, Vh Constant**
Measure Ia and Is as a function of Vg, stepping Va with Vs and Vh constant. This is the normal transfer curve from which the transconductance can be derived. In case of triode the screen terminal can be left un-connected.
2. **I(Vg, Va=Vs) with Vh Constant**
Measure Ia and Is as a function of Vg, stepping Va=Vs with Vh constant. This measurement type was especially added to measure the transfer curves of both sections of a double triode simultaneously. During this measurement the first anode is connected to the anode terminal of the uTracer, while the second anode is connected to the screen terminal. The screen power supply in this case acts as the anode supply for the second anode. By plotting Ia along the left y-axis and Is along the right y-axis, both anode currents can be displayed in one plot.
3. **I(Va, Vg) with Vs, Vh Constant**
Measure Ia and Is as a function of anode voltage, stepping the grid voltage, with the screen voltage and heater voltage constant. This is a normal output curve measurement. From the plot the output or plate resistance can be extracted.
4. **I(Va, Vs) with Vg, Vh Constant**
Measure Ia, Is as a function of anode voltage, stepping the screen voltage with the grid and heater voltage constant.
5. **I(Va=Vs, Vg) with Vh Constant**
Measure Ia, Is as a function of anode voltage = screen voltage stepping the grid voltage with the heater voltage constant. This measurement type is used to simultaneously measure the output characteristics of both sections of a double-triode. During this measurement the first anode is connected to the anode terminal of the uTracer, while the second anode is connected to the screen terminal. The screen power supply in this case acts as the anode supply for the second anode. By plotting Ia along the left y-axis and Is along the right y-axis, both anode currents can be displayed in one plot.
6. **I(Vs, Vg) with Va, Vh Constant**
Measure Ia and Is as a function of screen voltage, stepping the grid voltage with the anode and heater voltage constant.
7. **I(Vs, Va) with Vg, Vh Constant (+Vg mode)**
This measurement type is used to measure the transfer curve of a triode in positive grid bias mode! In this case the screen power supply is connected to the grid, and the grid terminal of the uTracer is left unused. As a result the anode and grid bias curves are recorded as a function of positive grid bias stepping the anode voltage.
8. **I(Va, Vs) with Vg, Vh Constant (+Vg mode)**
This measurement type is used to measure the output curve of a triode in positive grid bias mode! In this case the screen power supply is connected to the grid, and the grid terminal of the uTracer is left unused. As a result the anode and grid bias curves are recorded as a function of positive grid bias stepping the anode voltage.

9. **I(V_h, V_g) with V_a, V_s Constant**

Measure I_a and I_s as a function of heater voltage, stepping the grid voltage with the anode and screen voltages constant. Since in general it will take some time for the heater to stabilize after a new bias point is set, a delay may be inserted between the application of the new bias point and the actual measurement. The delay can be specified in the text field marked “Delay” in the lower left corner of the form.



10. **I(V_h, V_a) with V_g, V_s Constant**

Measure I_a and I_s as a function of heater voltage, stepping the anode voltage with the grid and screen voltages constant. Since in general it will take some time for the heater to stabilize after a new bias point is set, a delay may be inserted between the application of the new bias point and the actual measurement. The delay can be specified in the text field marked “Delay” in the lower left corner of the form.

3.3 – Averaging and Ranging

To cover the complete measurement range of 0 to 200 mA with sufficient accuracy, the uTracer uses a Programmable Gain Amplifier (PGA) in both the anode- as well as the screen current measurement circuits. By selecting the proper gain factor, the complete measurement range is divided into 8 sub-ranges from 0-1 mA to 0-200mA which are all recorded with 10 bit resolution. Additionally, for measurements in the low current regime it is necessary to perform several measurements to reduce noise.

In normal use a special algorithm takes care of the automatic switching between ranges so that the most optimal resolution is achieved. Depending on the measurement range the algorithm will also determine the number of measurements to be averaged. For most measurements the automatic ranging and averaging works just fine, however in some cases it may be better to select the measurement ranges or averaging manually. This can be done by selecting the required measurement ranges or averaging from a drop-down menu on the measurement form.

Some examples of when you would like to consider setting ranges/averaging manually:

- If only a quick impression of the tube characteristics is required, the measurement can be speeded up by setting the averaging to “none”.
- Normally the algorithm will try to measure both the anode and the screen current with the same accuracy. This implies that if a triode is measured and no screen is connected, or when a pentode is measured in the regime where the screen current is very low, the algorithm will select the lowest range for the screen current combined with high averaging to reduce the noise. The extra averages delay the measurement while they are not needed for the anode current. To speed up the measurement the range for the screen current can be manually set to 0-200 mA. In this case the algorithm will determine the number of averages only on the value of the anode current.
- Although the gains of the PGAs are quite accurate, it is still possible that small deviations from the ideal gain in different ranges may cause small dips or bumps in the graph when the algorithm changes the range in the middle of a graph. For the nicest “show pictures” it may therefore be best to set the range manually (Fig. 3.3).
- In the “Quick Test” the derivatives (R_p and G_m) are determined from just two points on the curve. Any noise on these two points will directly result in variations in R_p and G_m. This is especially true when the gradients are small (R_p is high, e.g. pentode). In this case it is better to make sure that the algorithm does not switch ranges from one point to the other. Additionally, regardless of the current level, it is sensible to manually set the number of averages to e.g. 16X. The best approach is to do a first quick test with ranging and averaging set to automatic just to find current levels and then to select the proper range manually, set average to 16X and redo the test.

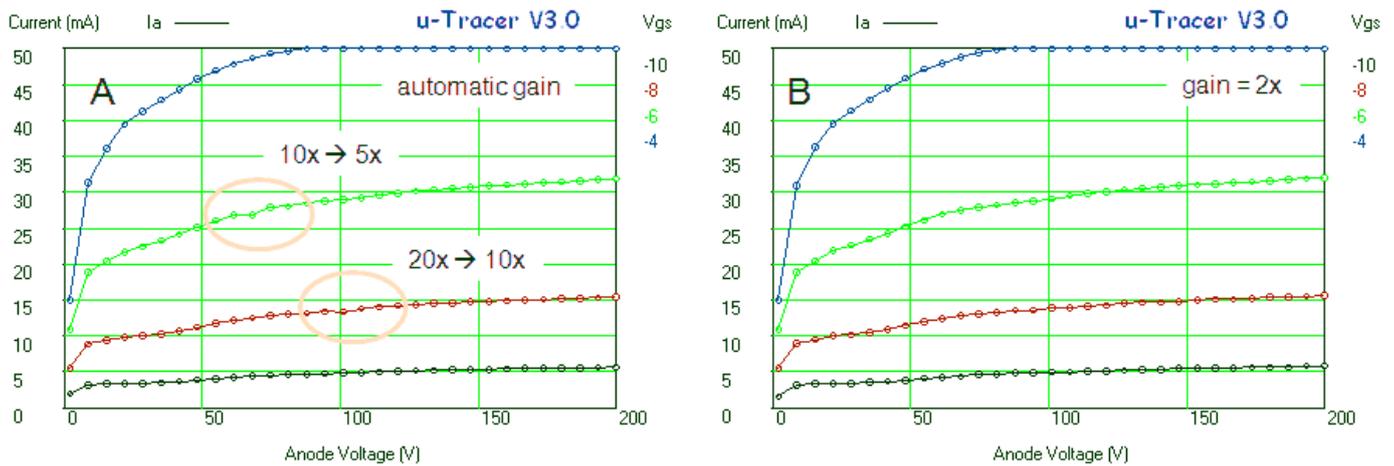


Figure 3.3 An $I_a(V_a)$ plot for $V_g = -10V, -8V, -6V, -4V$ and $V_s = 200V$. The left set of curves was measured using the automatic gain option. The automatic selection of another gain setting causes small discontinuities in the current. Re-measuring the set of curves with a fixed gain results in a smooth set of curves.

3.4 – Compliance

No feature on the GUI has been so hard to explain as the compliance setting. To explain what it is, let's consider a short circuit or the situation when a very heavy load is applied to the output. The first thing that will happen is that immediately the current limiting circuit around the high voltage switch will kick in, limiting the current through the tube to about 240 mA. It can only do so by biasing the transistor of the high voltage switch somewhere in between full conduction and complete cut-off. As a result almost the entire voltage will be dropped over the transistor resulting in an enormous dissipation! Suppose for example that the reservoir capacitor was charged to 300 V, then a short circuit will cause an instantaneous dissipation in the high voltage switch of about $300 \times 0.24 = 75 \text{ W}$!

It will be clear that the high voltage switch will not like such a huge dissipation! It is therefore important to switch off the current in the circuit as quickly as possible. This is done by the second line of defense against short-circuits namely the processor itself. When the current increases beyond the value specified by the compliance setting of the GUI, an interrupt will be generated that will terminate the measurement pulse. This takes about $20\mu\text{s}$, which in tests during the development of the uTracer, appeared to be short enough to prevent damage to the transistor.

By setting the compliance to a certain value, it is not the maximum current that is set, but rather the current level above which the processor will shut down the circuit. Thus setting the compliance to a level that is lower than 200 mA provides increased protection against overload conditions. Admittedly its use is limited, but it was an option that came absolutely for free.

3.5 – Delay

Normally when a point is measured, the GUI sends a command to the uTracer which tells it to set the required voltages and when they are reached to issue a measurement pulse. When a "Delay > 0" is specified, a delay is inserted in between the setting of the voltages and the measurement pulse. The delay is used for:

1. Measurements which use the heater voltage as running (x-axis) variable. Obviously the heater will need some time to reach the equilibrium temperature corresponding the specified voltage. For directly heated tubes the delay can be in the order of seconds, while for indirectly heated tubes something in the order of 30 seconds is more appropriate.
2. [Continuous measurements](#) e.g. for testing Magic Eye tubes. A delay of a few seconds makes it possible to better observe the tube.

4 – Graphical Output

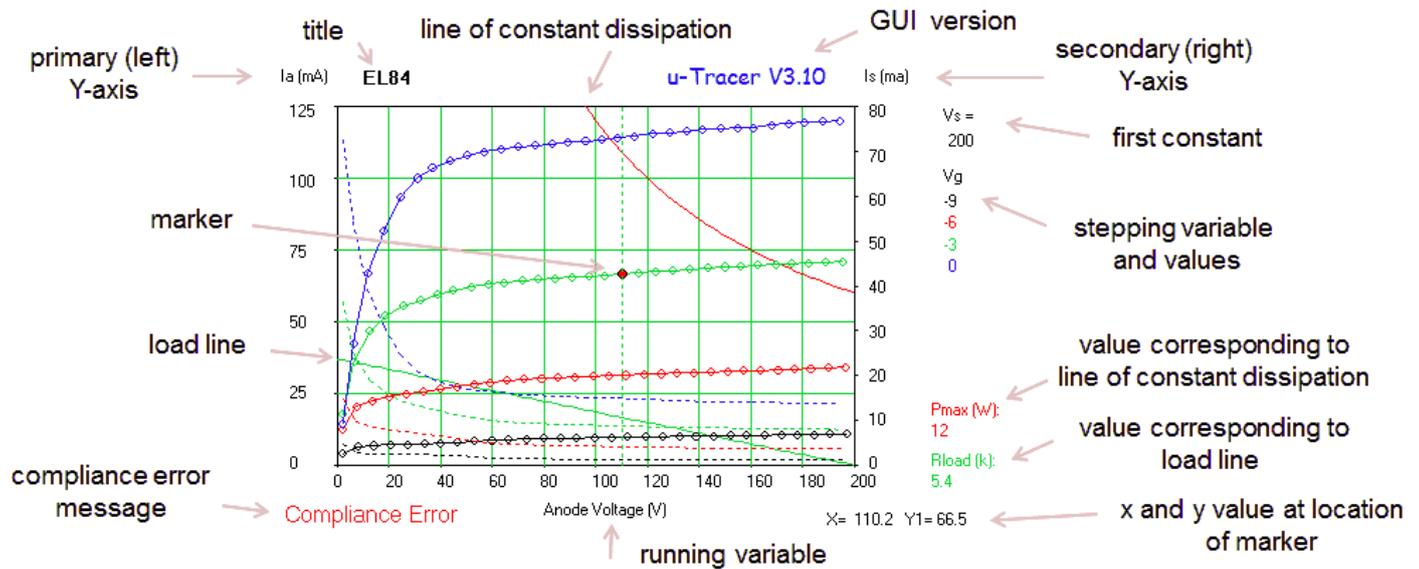


Figure 4.1 The graphical output control section

Figure 4.1 shows the main features of the graphical output control section. In this example the output characteristics of an EL84 were plotted for three different grid voltages. The primary (left) y-axis is used for the anode current, while the secondary (right) y-axis is used for the screen current. Most of the objects in the graph are obvious or intuitive, but a few objects will benefit from some further explanation.

It is possible to position a marker on the plot by positioning the mouse on the required spot and left clicking the mouse. The GUI will draw a vertical line, and jump to the line closest to the mouse pointer. This can be a line belonging to the left- or the right-axis. The X and Y values corresponding to the marker position are shown on the lower right-hand corner.

Apart from the measured curves, the user has the possibility to add a load line and a line of constant dissipation. The load resistance and the dissipation values can be entered in the proper fields in the input section below the graph. Both lines are just there for convenience of the user and *in no way further interact with the working of the uTracer*. When the maximum allowed dissipation for the tube is entered, the line of constant dissipation will show the Safe Operating Area (SOA) of the tube. In normal continuous mode, a tube may not be operated beyond this line. However, because the uTracer works in pulse mode, there is no problem at all in measuring the tube characteristics beyond the SOA are!

Apart from the anode- and screen-currents the user has the choice out of seven other parameters to plot in the graph. These purpose of these parameters will be explained in section 4.2. In the next section the other controls related to the graphical output will be discussed.

4.1 – Controlling the Graphical Output

Normally after a measurement type is chosen in the measurement section, the graphical section is initialized in such a way that the way that automatically a representable graph is obtained of the anode current versus the running variable without the need to change or set any of the controls. By default all axes are automatically scaled, or track the Start and Stop values of the running variable. Nevertheless, there are many options to adjust what is plotted, or how it is plotted.

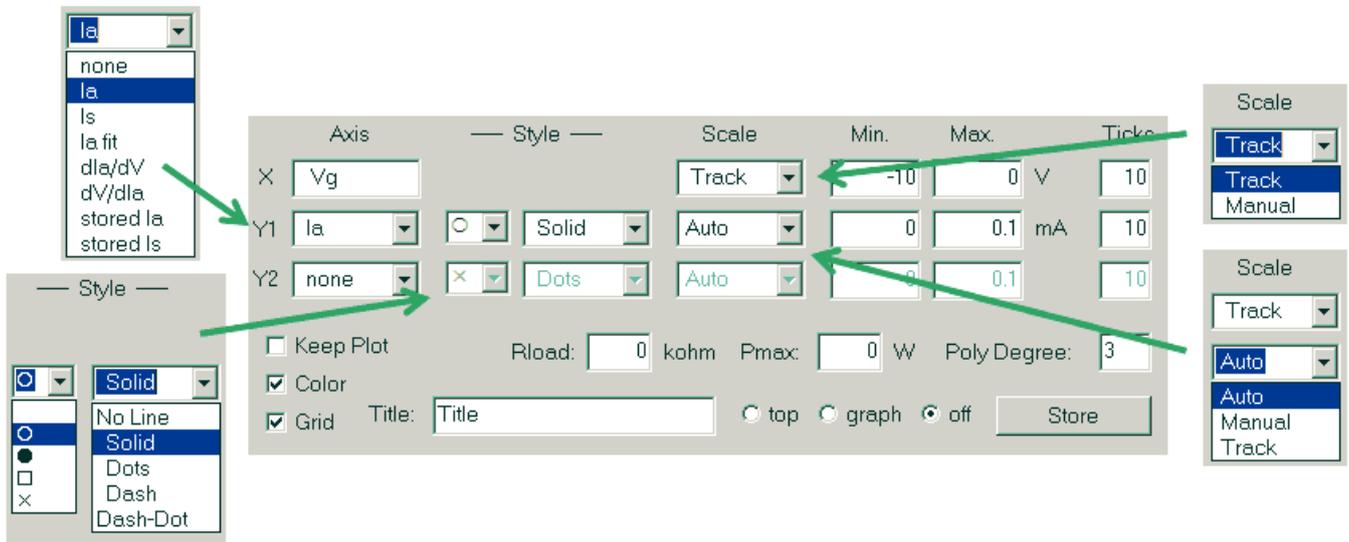


Figure 4.2 The controls of the graphical output section

At the heart of the graphical output control section are three lines with drop-down menus and text fields. The three lines control what is plotted along the three axes of the graph, how the data is plotted and how the axes are arranged. From top to bottom the three lines control the X-axis, the Y1-axis and the Y2-axis. What is to be plotted along the X-axis is determined by the measurement type. What is to be plotted along the two Y-axes can be chosen from two drop-down menus (Fig. 4.2). Most obvious are Ia and Is. The meaning of the other five variables will be explained in more detail later. By default only the primary (left) axis is used to display the anode current. The secondary (right) axis can be switched on by selecting any of the seven parameters from the Y2 drop-down menu.

The other controls in this area are quite self-explanatory: the second and third column of drop-down menus, under the heading “— Style—“ determine the type of marker used and the line style.

The choices in the drop-down menus under the heading “Scale” perhaps need some explanation. There are three choices: “Auto”, “Manual”, and “Track”. In “Auto” the scaling of the axis is automatic and depends on the largest value measured. There is a special algorithm which ensures that sensible and “round numbers” are used along each axis. Using the “Track” setting, the Min and Max values of the axis are determined by the values chosen in the measurement section. When “Manual” is selected, the axis is scaled according to the values specified in the fields of the columns marked “Min” and “Max.” The number in the column marked “Ticks” specifies the number of sub-divisions on the axis.

The user has the option to add a title either above or in the plot by entering the text in the field labeled “Title” and by ticking one of the radio-buttons “top” or “graph”. The function of the boxes marked “Color” and “Grid” are straightforward. The values entered in the boxes labeled Rload and Pmax respectively determine the locations of a resistive load line and a line of constant dissipation on the graph. As mentioned, these in no way affect the working of the circuit, and are just there for the convenience of the user. Entering a zero value in one of the fields will remove the corresponding line from the graph. The other features of the graphical output section will be discussed in more detail in the next sections.

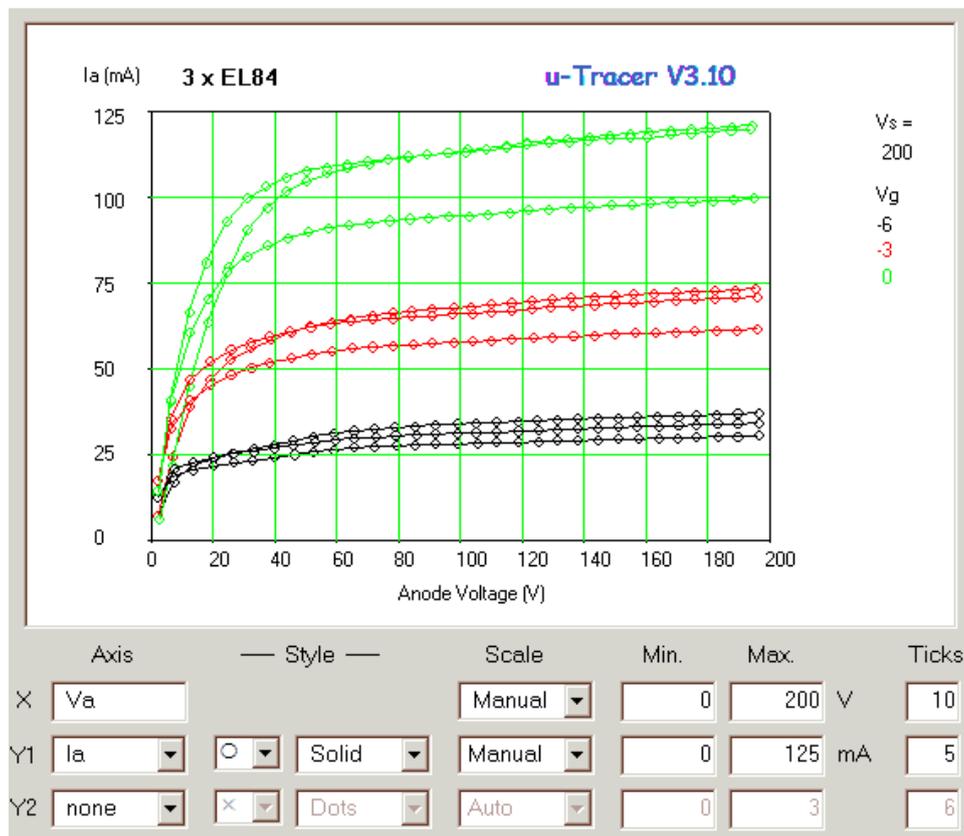


Figure 4.3 Three EL84's compared using the "Keep Plot" option. Note how the scaling of the axes has been set to manual.

4.2 – Keep Plot

The "Keep Plot" option has been added as a crude method of comparing the characteristics of several tubes in one plot. The working is very simple, but the proper use of this feature requires a bit of practice. During normal operation of the uTracer, any existing graph is erased every time a new object or measurement is added to the plot. When the "Keep Plot" box is ticked this erasure is omitted so that objects can only be added to the graph. The "Keep Plot" option does not work in combination with the autoscaling of the axes, because every time one of the axes is redrawn it will be added to all existing ones, resulting in an enormous mess (try it for yourself). The proper use of the "Keep Plot" option is as follows:

1. Make an orienting plot of the first tube using the standard auto-scaling of the axes.
2. Switch the scaling of the axes to manual, and adjust the Min and Max values if needed
3. Tick the "Keep Plot" box
4. Remove the first tube from its socket and insert the next one. There is no need to first switch of the heater supply and then to switch it on again (that is if you are not too squeamish about the heater).
5. Perform a new measurement. The new graph will now be added to the first one.
6. Insert, measure, and display results for additional tubes as needed

Figure 4.3 by way of illustration shows a comparison in one plot of the output characteristics of three EL84's I had lying around

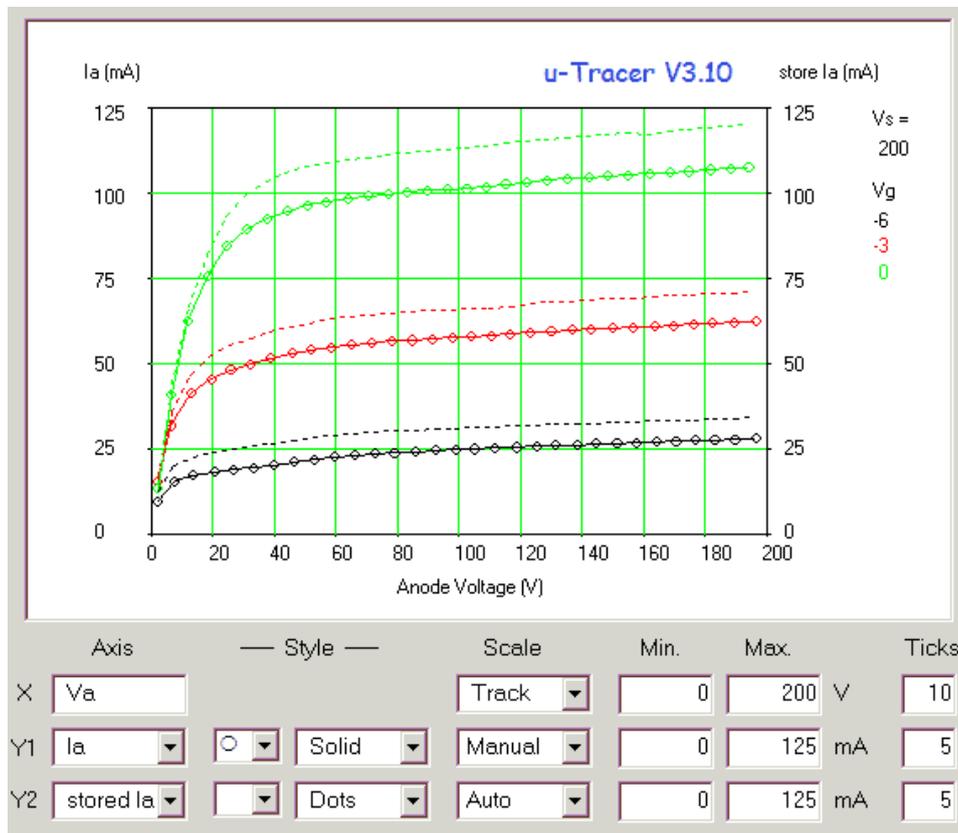


Figure 4.4 In this graph the output characteristics of an EL84 with heater voltage 5 V (solid lines with markers, left axis) is compared to the stored reference curves measured at a heater voltage of 6.3 V (dashed lines, right axis).

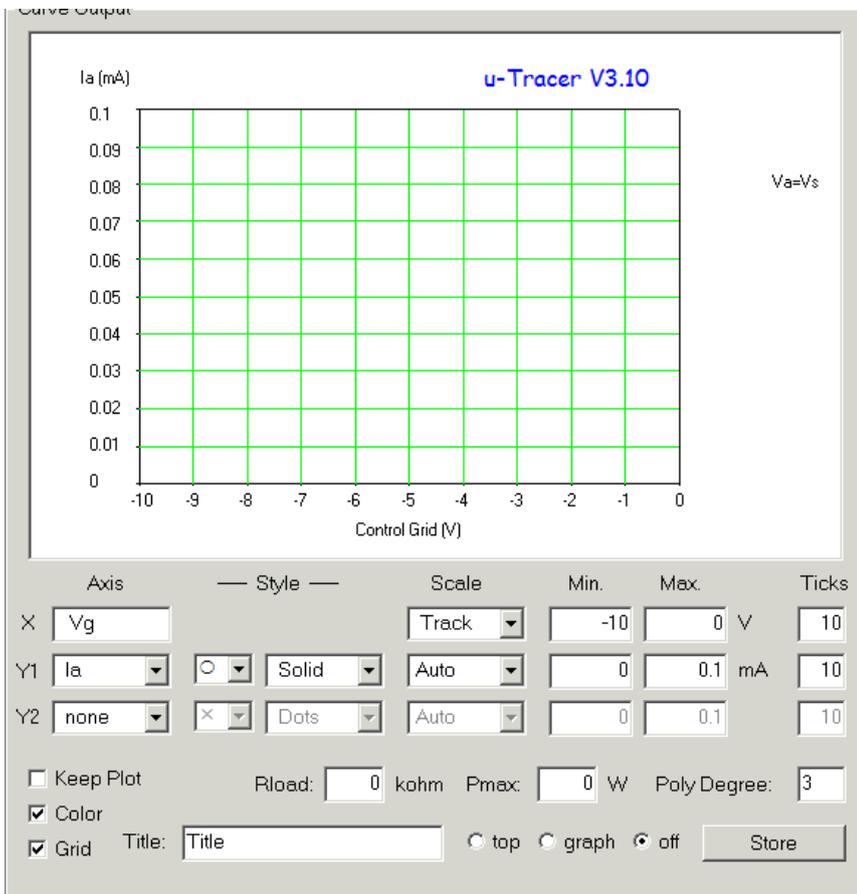
4.3 – Matching of tubes (Storing a Graph)

It is possible to store the complete data of an entire measurement to RAM memory so that it can be retrieved at a later stage e.g. to investigate the matching of two tubes. This option is not to be confused with the “Save Plot” or “Save Data” functions which save the plot or data to a file for further processing by other programs.

Pressing the command button labeled “Store” causes the current plot to be copied to memory. After the data has been stored it can be retrieved by selecting “stored Ia” or “stored Is” in the drop-down menu underneath the heading “Axis” in the graphical output control section (Fig. 4.2). The stored values can be plotted against the secondary Y-axis to enable easy comparison to a new graph. In Fig. 4.4 the “Store” option was used to compare two sets of curves measured at different heater voltages.

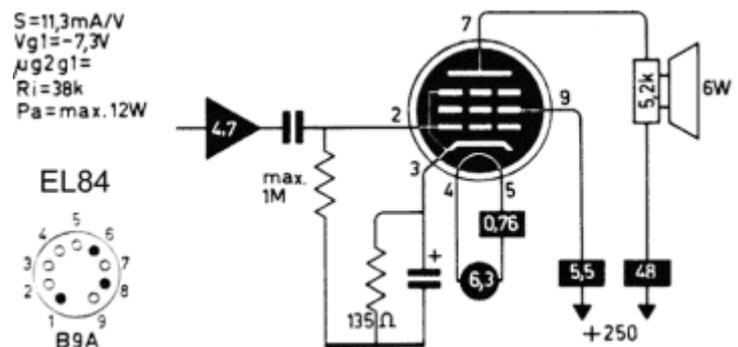
4.4 – Transconductance and plate-resistance

The transconductance of a tube is the variation in anode current as a result of a small variation in grid voltage at a given bias point. In other words, it is the slope of the anode current versus grid voltage at that bias point. In mathematical terms, the transconductance is the first derivative of the anode current with respect to the grid voltage.



This simplest way to measure the transconductance is to draw a line through two consecutive data points on the $I_a(V_g)$ curve. The transconductance is then nothing more than the slope of this line. However, differentiating a measured curve this way usually results in a very noisy transconductance curve as a result of noise in the measured data. This is especially true when the gradient of the curve is very small. In the version 3 uTracer therefore a better approach is used: After measurement of a set of $I_a(V_g)$ curves, a polynomial is fitted through the measurement points. This polynomial smoothes out the fluctuations in the measured data. The order of the polynomial is chosen such that the polynomial accurately represents the measured data. After the fit the polynomial is analytically differentiated resulting in a perfect - noise free - transconductance curve [\[MORE\]](#).

The figure to the left shows in an animation the step-by-step the extraction of the transconductance curves from a set of $I_a(V_g)$ curves of an EL84 ($V_a=V_s$). After the measurement of the $I_a(V_g)$ curves the line is switched off so that only the markers remain. Next the fitted curve is selected to be plotted against the right axis. The optimal order of the polynomial is determined experimentally, and usually something between 3 and 6 gives a good result. The order of the polynomial should not be too high, however, because that can result in oscillations in the transconductance curve. When the fit is ok, the derivative of the current to the voltage (dI_a/dV) can be plotted. An alternative way to find the optimum polynomial degree is to directly plot the transconductance, and then to vary the polynomial degree. At first increasing the order of the polynomial should result in a significant change in the transconductance curves. After a certain point the curves will not change much when the order is increased. As the order is increased further, oscillations in the transconductance curves will begin to appear. The optimum is somewhere in the range where the transconductance is relatively independent of the degree of the polynomial. Finally, the marker is seen to exactly match the transconductance at the bias point recommended in my "Muiderkring" tube handbook, 11.6 mA/V! (see inset, $V_g = -7.3$ V and $V_a = V_s = 250$ V).



The same procedure can be used to extract the output or plate resistance from a set of output curves. In this case the inverse of dI_a/dV is plotted: dV/dI_a . Extracting the output resistance of a triode is relatively straightforward since the gradient of the curves is usually relatively high. The left graph in Fig. 4.5 shows the extracted output resistance of an EL84 connected as triode. The curves are nice and smooth, apart from some noise in the low current regime. In normal pentode mode the extraction of the plate resistance is more difficult. The fitted anode current curves look ok, but they contain small fluctuations that show up as oscillations in the plate resistance. It doesn't require a lot of imagination to conclude that the plate resistance is somewhere in the range of 25-30 kohm. In this case an 11th order polynomial was found to be optimal.

Note that a conventional extraction of the plate resistance from the tangent of a line drawn through two neighboring measurement points would result in a much noisier plot.

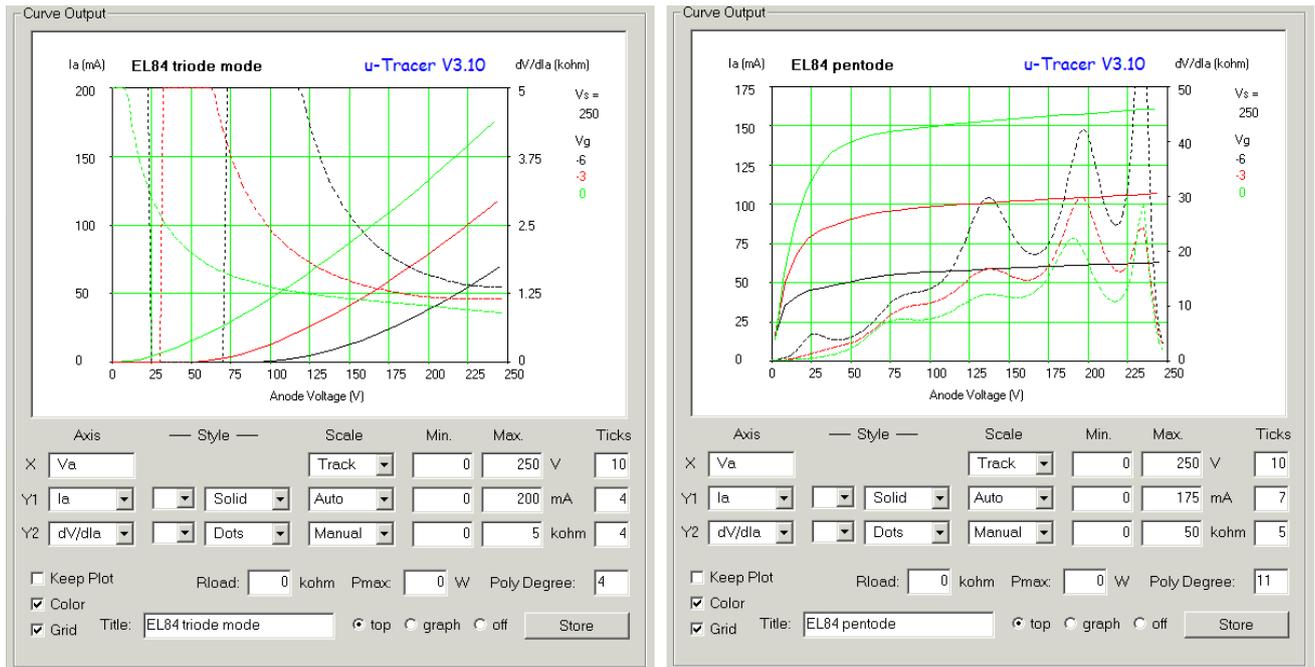


Figure 4.5 Output resistance versus anode voltage (dotted line) for an EL84 switched as triode (left) and pentode (right).
Number of averages 16 with anode range fixed to 0-200 mA.

5 – Quick Testing

Using the “Quick Test” option it is possible to quickly determine the most important parameters of a tube (nominal currents, gm, Rp and mu) without having to plot a complete set of curves, or having to go through the complete curve fitting procedure. Starting from GUI version 3p10 the “Quick Test” has been added as an additional feature to the GUI. In this section we will review how it works, what is measured, and how the Quick Test option can be used.

5.1 – Quick Test introduction

Most quick reference tube manuals don’t give complete sets of curves for tubes, but only cite the most important parameters in an optimal bias point. If we take the EL84 as an example, then the optimal bias point is given as Va=250 V, Vs=250 V, and Vg = -7.3V, and the most important parameters in that point are Ia=48 mA, Is=5.5 mA, gm=11.3 mA/V, Rp=38 kohm. We distinguish between directly measurable parameters, and indirectly measurable parameters. Two parameters which can be measured directly at the given bias point are the anode and the screen currents. The plate resistance, transconductance, the amplification (mu) cannot be measured directly, but are derived from the tube’s characteristics at the specified bias point. The transconductance for example is the first derivative of the anode current with respect to the control grid voltage ($\partial I_a / \partial V_g$) at the specified bias point, and the plate resistance is the inverse of first derivative of the anode current with respect to the anode voltage ($\partial V_a / \partial I_a$) at the bias point.

For a triode we have only the anode current as function of the grid and the anode voltages, or $I_a(V_g, V_a)$, so there are only two derivatives: the transconductance (gm) and the plate resistance (Rp). The amplification factor mu can simply be calculated from $\mu = g_m * R_p$. For a tetrode or pentode, the situation is much more complex. Here we have to deal with the anode current (Ia) as well as the screen current (Is), which in principle are functions of all three terminal voltages so: $I_a(V_g, V_a, V_s)$ and $I_s(V_g, V_a, V_s)$. Now there are no less than six derivatives which all have a physical meaning and which all may be relevant for the design of a circuit:

	I_a	I_s		I_a	I_s
V_a	$\frac{\partial I_a}{\partial V_a}$	$\frac{\partial I_s}{\partial V_a}$		$R_p = \frac{\partial V_a}{\partial I_a}$	$\frac{\partial I_s}{\partial V_a}$
V_s	$\frac{\partial I_a}{\partial V_s}$	$\frac{\partial I_s}{\partial V_s}$		$\frac{\partial I_a}{\partial V_s}$	$R_s = \frac{\partial V_s}{\partial I_s}$
V_g	$\frac{\partial I_a}{\partial V_g}$	$\frac{\partial I_s}{\partial V_g}$	\Rightarrow	$g_m = \frac{\partial I_a}{\partial V_g}$	$\frac{\partial I_s}{\partial V_g}$

Figure 5.1 Possible derivatives for a pentode

Some of these derivatives are more naturally presented as a *resistance* such as the plate and screen resistances, while the others are better presented as *transconductances*. In the “standard” GUI the derivatives are obtained by first measuring the complete curve, fitting a polynomial to the curve, analytically differentiating the curve and finally plotting the derivative. Although this method produces nice smooth curves as a function of bias, it is in most cases sufficient to know the derivatives at a particular bias point. In this case the derivative can be *approximated* by drawing a straight-line through two points $V_o - \delta V$ and $V_o + \delta V$, where V_o is the bias point and δV is a small offset. As an example the transconductance can be approximated with:

$$g_m = \frac{I_a(V_g + \delta V_g, V_a, V_s) - I_a(V_g - \delta V_g, V_a, V_s)}{2\delta V_g}$$

The smaller δV is chosen, the more accurate the approximation will be, but also the more sensitive the result will be to noise in the measured data. In principle now only six measurements are required to calculate all the derivatives for a pentode as shown in Figure 5.1, and a seventh measurement gives the currents in the bias points itself. For a triode only five measurements in total are needed.

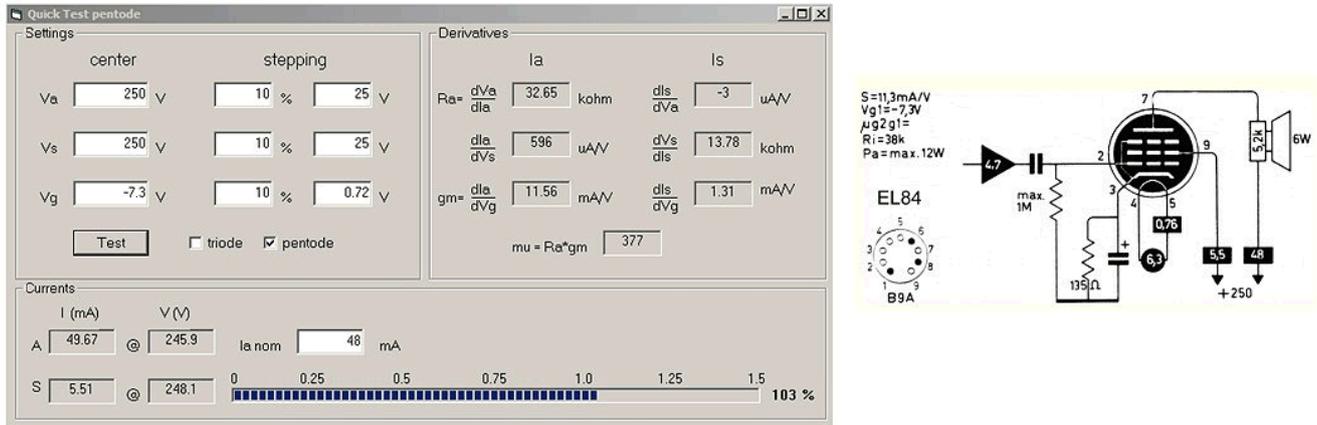


Figure 5.2 Quick Test of an excellent EL84 (6BQ5)

5.2 – Using Quick Test

The “Quick Test” option which measures all the derivatives in a specified bias point has been implemented as an alternative to a full curve characterization. The Quick Test is started by clicking the corresponding command button in the measurement section. Note that the heater must be activated to perform a Quick Test measurement.

The use of the Quick Test option is pretty straightforward:

1. Connect a tube to the uTracer, and start the GUI
2. Set the desired heater voltage. There is no need to select a particular measurement type, or to set any other bias value in the main measurement form since these are set in the Quick Test form. However, the “Range”, “Compliance” and “Average” selections retain their original functions, and the results of the Quick Test may depend on their settings. Using the default values is usually a good starting point.
3. Switch on the heater in the normal way
4. Click “Quick Test.” This will open the Quick Test form;
5. Select triode or pentode (=tetrode) depending on what type of tube is tested
6. Enter the desired bias point values in the appropriate fields. By default the GUI uses δV 's which are 10% of the bias value inputs. The δV 's may be increased or decreased for each input as required by entering either a new percentage value or by entering a δV directly. In both cases the other fields will be updated automatically.
7. When the nominal anode current at the bias point is known, it can be entered in the appropriate field. At the end of the test, this value is used to produce a bar graph that shows the anode current relative to the nominal value along with a calculated percentage.
8. Start the test by pressing “Test.” A triode test will take approximately five seconds, while a pentode test will take about eight seconds

All of the Quick Test bias values and settings are stored when the “Save Settings” function in the main GUI form is executed.

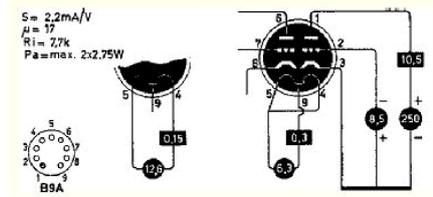
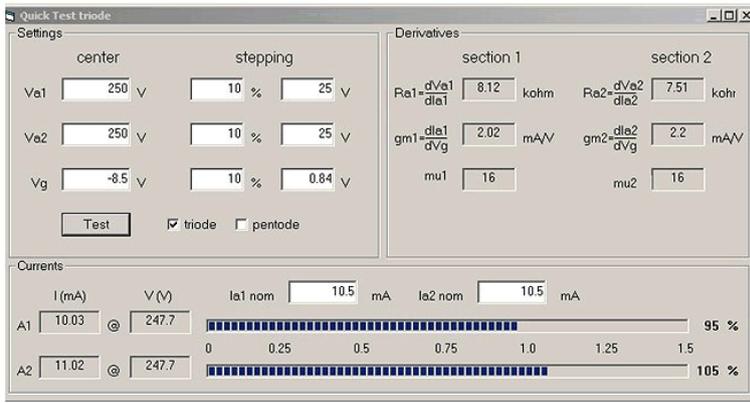


Figure 5.3 Quick Test of an ECC82 (12AU7) double triode. The plate resistance (R_p), transconductance (g_m) and voltage amplification (μ) are simultaneously measured for both sections.

Tips and things to keep in Mind:

1. The triode Quick Test has been designed so that both sections in double-triodes are tested simultaneously (Fig. 5.3). In this case the anode of the second section is connected to the screen terminal of the uTracer. When single triodes are tested the fields for the second triode can simply be ignored.
2. Plate and screen resistances larger than 1M ohm are displayed as “>1M”. Transconductances larger than 200 mA/V are displayed as “>200”.
3. To test triodes in A2 mode (positive grid bias), the pentode Quick Test is used and the grid is connected to the screen terminal of the uTracer. In this case $\partial I_a / \partial V_s$ gives the transconductance while $\partial V_s / \partial I_s$ gives the grid impedance!
4. The measured currents used to calculate the derivatives can be stored and viewed by storing the Measurement Matrix: press “Save Data” followed by “Save Measurement Matrix.”
5. Especially for high plate resistances (pentodes/tetrodes), the measured value can be significantly affected by a tiny current fluctuations. It can therefore be better to fix the measurement range to a certain value to avoid gain switching in between two measurement points and to manually set the averaging to a high value.
6. When a δV value is entered that would result in a bias setting beyond the capabilities of the uTracer, the value is adjusted to stay within the maximum and a warning message is generated.

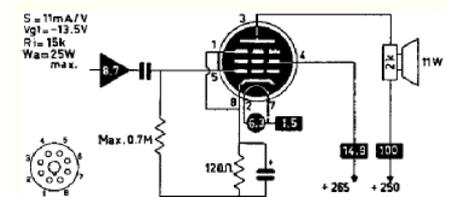
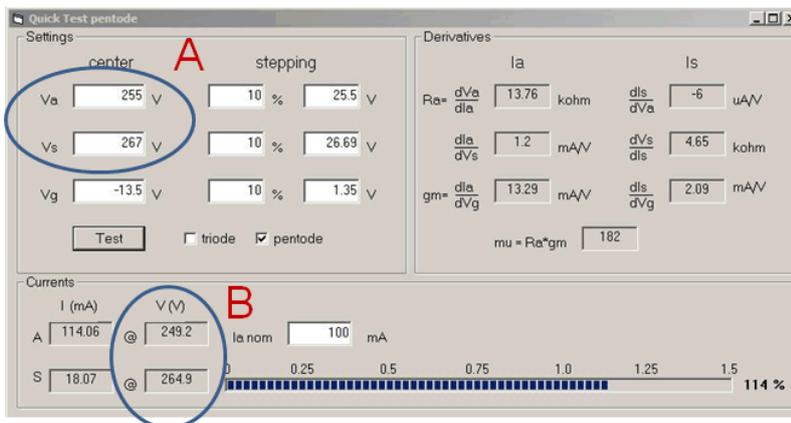


Figure 5.4 Quick Test of an excellent EL34 (6CA7). To obtain the desired bias voltages: $V_a=250$ V, $V_s=265$ V (circle B) it may be necessary to increase the set point values slightly (circle A).

6 – Saving Plots and Measurement data

In the main form's "Miscellaneous" section there are a number of command buttons which are used to store the plot or the measured data to a file, or to save and retrieve the complete measurement set-up. These options are explained below. Note that in all cases the default location for the files is the folder into which the uTracer was installed, but any other folder may be specified.

6.1 – Saving a Plot

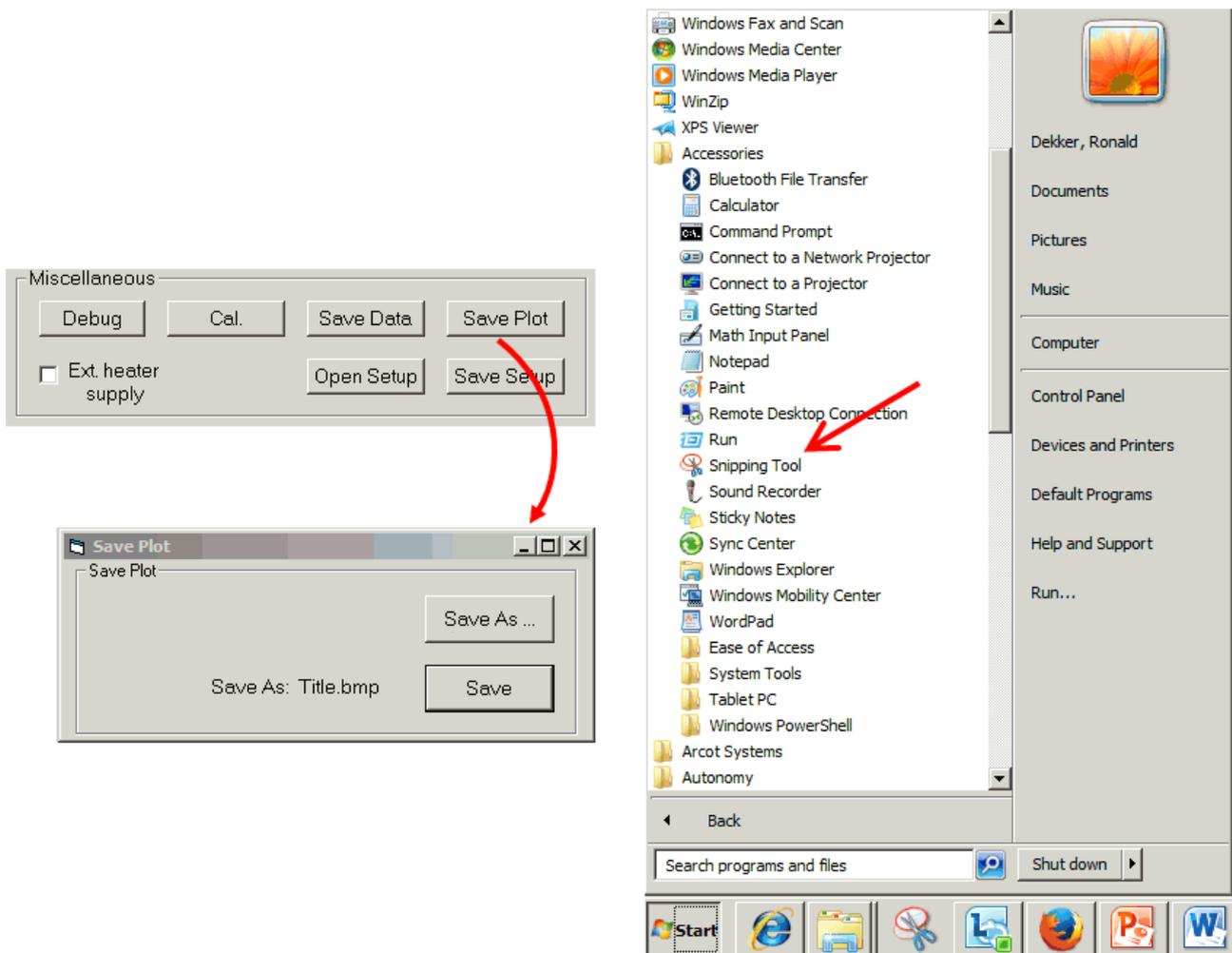


Figure 6.1 Left: plotting the graph to a bmp file. Right: location of the snipping tool which can be used to grab the graph from the screen.

A limitation of Visual Basic, the programming language of the GUI, is that it can only save graphics in bit map (bmp) format. Personally I do not think that is a problem, since bit map format graphics can easily be imported into standard programs like Word and PowerPoint. On clicking the "Save Plot" button the small form shown in the left half of Fig. 6.1 opens. By clicking "Save As", A Dialog Control opens which allows the user to browse for a particular folder and to specify a file name. The extension will be .bmp

Personally I prefer to use the "Snipping Tool" which is a standard Windows tool that can be found under "Accessories" (Fig. 6.1 right). I even have this tool permanently copied to the taskbar at the bottom of the windows screen. It can be used to grab an arbitrary part of the screen and write it to a file in .jpg or .gif format, or to copy the selected graphic directly into Word or PowerPoint. In fact most of the images in this manual were made by first grabbing a graphic from the screen, copying it to PowerPoint for editing (adding

text, arrows or other pictures), and then grabbing the final image and writing it to a file.

6.2 – Saving measurement data

Very likely the user will want to save the measurement data to import it into another application or program, such as e.g. Excel. There are a number of ways that the measured data can be written to a file, and in this section the different possibilities and formats will be discussed. Importation of the data into Excel will also be explained. A very simple measurement consisting of two curves with four points each (Fig. 6.2, right) will be used to illustrate the different file formats.

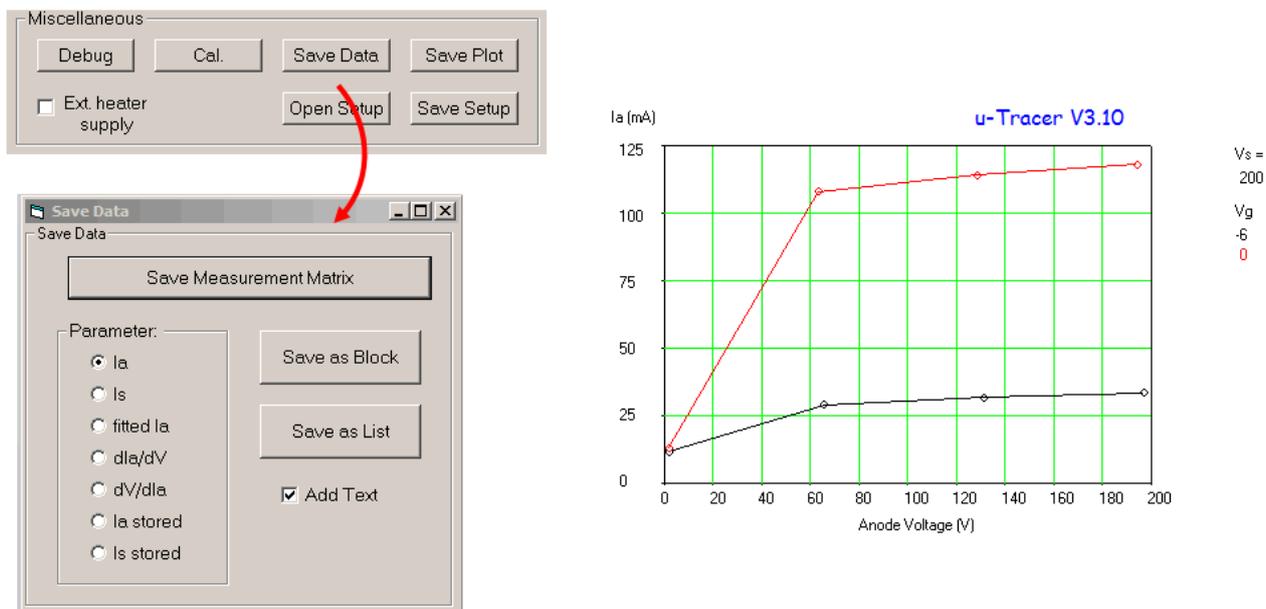


Figure 6.2 The “Save Data” form (left) and the test measurement used to illustrate the file formats in this section (right) .

The “Save Data” form is opened by clicking the “Save Data” button in the “Miscellaneous” section in the main form (Fig. 6.2 left). There are three formats in which the measured data can be saved: Measurement Matrix, Block, and List. The different formats are selected by clicking the corresponding command button. Both the points to be measured and the measured data are contained in the Measurement Matrix, an array used internally by the GUI. By pressing the “Save Measurement Matrix” button, the contents of this array are written to the specified file. This file, and all measurement data files, will have the extension .utd (**uTracer data**).

Point	Curve	Ia (mA)	Is (mA)	Vg (V)	Va (V)	Vs (V)	Vf (V)
1	1	11.75	15.1	-6	1.89	198.69	6.29
2	1	28.94	4.74	-6	65.61	198.19	6.29
3	1	31.65	3.95	-6	131.29	198.21	6.29
4	1	33.54	3.73	-6	197	198.55	6.29
1	2	13.05	71.87	0	1.87	197	6.29
2	2	107.94	15.91	0	63.17	198	6.29
3	2	113.97	14.26	0	128.46	198.37	6.29
4	2	117.95	13.41	0	194.47	198.38	6.29

Figure 6.3 Example showing the format of the saved measurement matrix

Figure 6.3 shows the format of the saved Measurement Matrix. The data is stored in eight columns. The first column gives the measurement point number within each curve, while the second column gives the curve number. The third and fourth columns are the anode and screen currents in mA. The last four columns contain the grid, anode, screen, and filament voltages. Note that the grid and filament voltages displayed are the set point values, while the anode and screen voltages are actually measured.

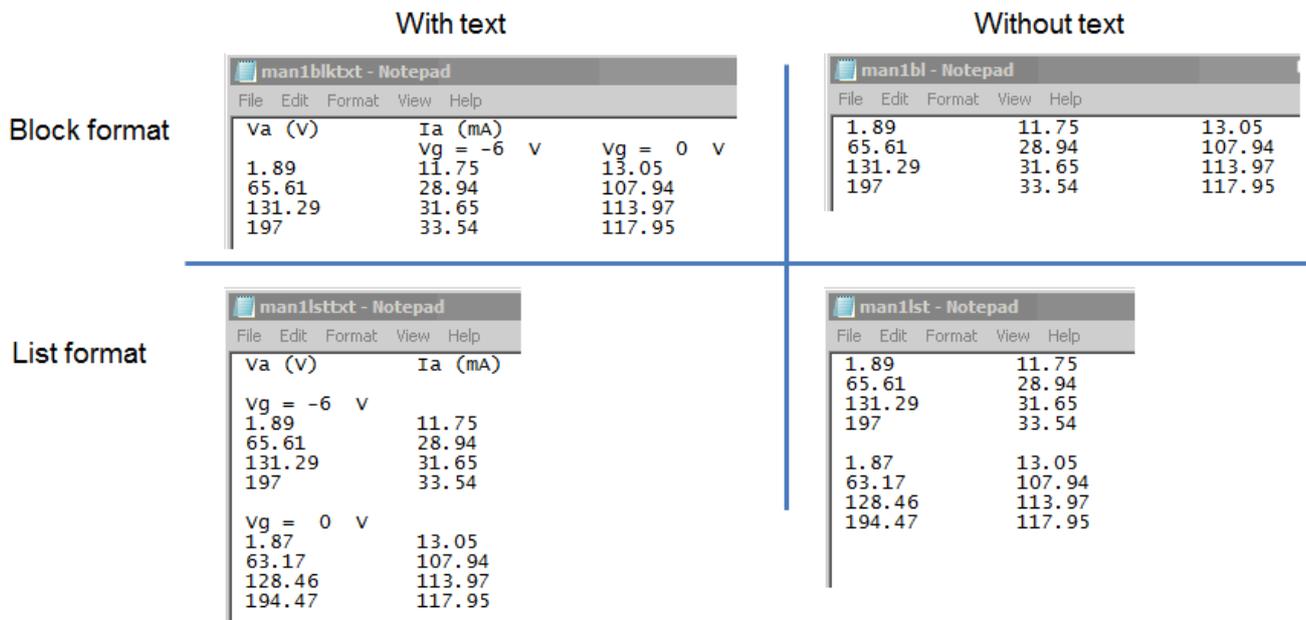
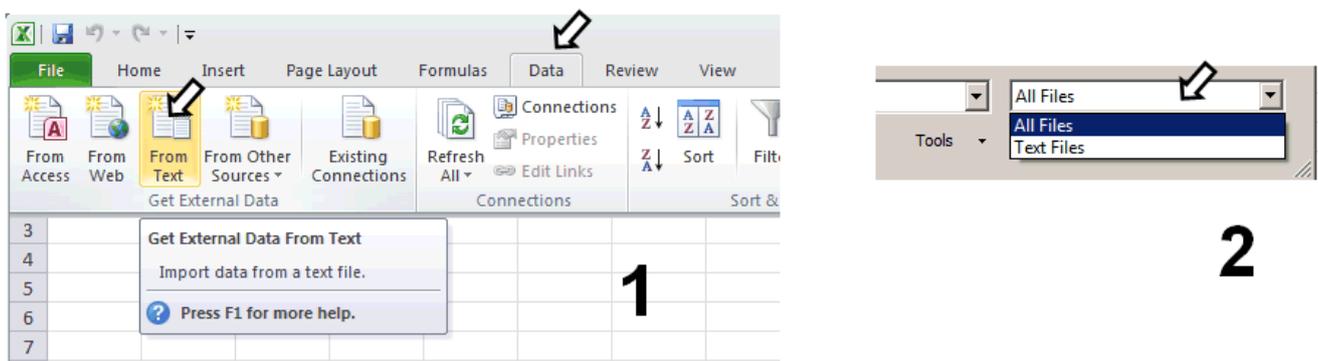


Figure 6.4 Examples of the “block” format (top row) and “list” format (bottom row). Both formats can be generated with and without text.

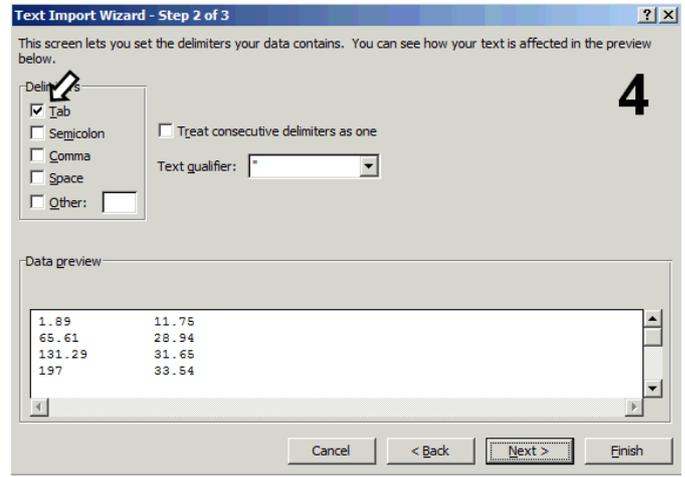
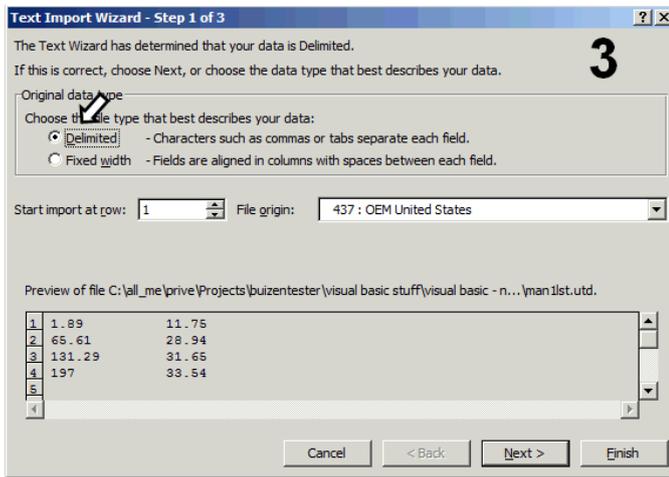
Figure 6.4 illustrates the “block” and “list” formats: In the “block” format the first column is the running (x-axis) variable of the first curve, and the following columns are the measured data for each curve. In the “list” format the data for both the running variable and the measured data for each curve are stored sequentially. Since the actual value of the running variable depends on the current, there can be a slight error in the current-voltage relationships in the block format. In the list format both the voltages and currents are stored in exactly the way that they are measured. To facilitate importing data into another application, the text in the files can be omitted by unselecting the tick box marked “Add Text” (Fig. 6.4). Instead of the anode current, any one of the variables in the “radio button” list on the form may be selected for saving.

6.5 – Importing data into Microsoft Excel

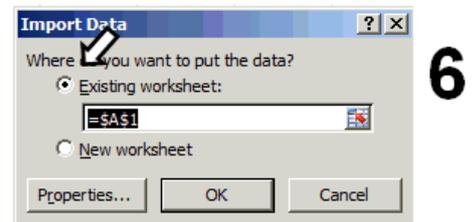
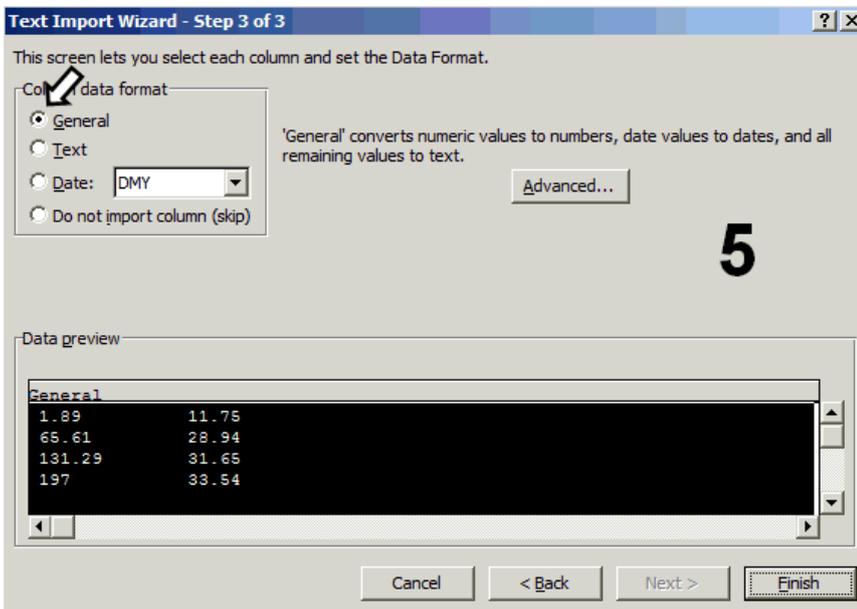
Importing data from a .utd file into Excel is very easy, and this section shows the process step-by-step using a file that has been saved with the “List” format. I am not particularly experienced with Excel, and it may be that there are simpler ways to import the data. For incidental use, this method works fine.



1. Under tab “data” select “From Text” from “Get External Data”.
2. In the file browser make sure to select “All Files”.



3. Select "Delimited" (default),
4. by "Tabs" (default)



	A	B	C
1	1.89	11.75	
2	65.61	28.94	
3	131.29	31.65	
4	197	33.54	
5			
6	1.87	13.05	
7	63.17	107.94	
8	128.46	113.97	
9	194.47	117.95	
10			

5. Select "General" for data format (default),
6. in "existing worksheet" (default);
7. Finished!

6.4 – Saving the Measurement Setup

One of the most useful features introduced in GUI 3p9 was the possibility to save all the measurement settings to a file so that they can be recalled later. By clicking the "Save Setup" command button in the Miscellaneous section, the user is prompted to enter a file name after which the complete measurement setup, pinning information, graphical output settings and Quick Test settings are stored to a file. This file is a standard ASCII file, which can be read with any text editor. The file has the default extension .uts (uTracer Setup). Note that to avoid decimal delimiter problems, all reals are stored as integers multiplied by 1000. A value of -1 is used to indicate the end of file, which makes it possible to extend the file in the future. Any saved measurement setup can be reloaded by clicking "Open Setup" followed by the selection of the particular setup file.

6.5 – The Calibration File

Small variations in component values make it necessary to calibrate the uTracer for optimal accuracy. To accomplish this, a set of variables is adjusted by means of a graphical “Calibration” form where slide bars are moved to set precise values for currents and voltages [\[MORE\]](#). The exact procedure to find the correct calibration values is described in the construction manual. The default value for each slide bar is 1.0 (center position), and each can be adjusted over a range from 0.9 to 1.1, or +/- 10%. The variables VaGain and VsGain adjust the anode and screen voltages while IaGain and IsGain adjust the measured currents. Vsuppl and Vgrid allow the exact supply and grid voltages to be set, and Vsat adjusts the voltage drop over the high voltage switch during the measurement pulse.

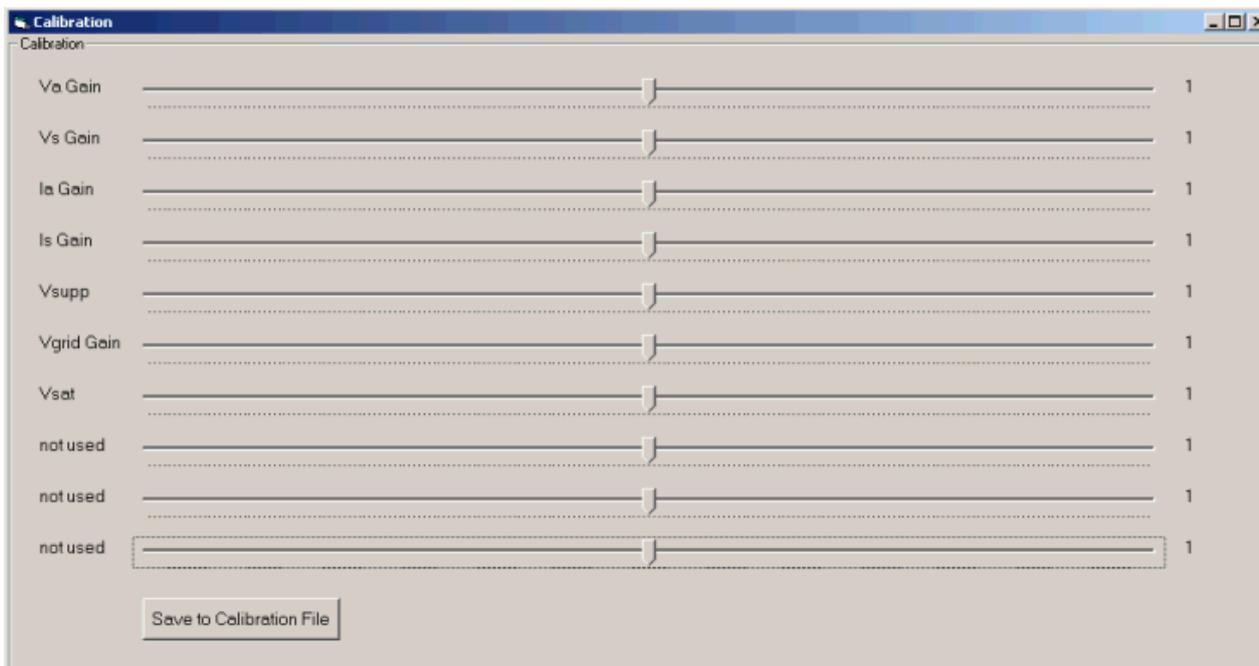


Figure 6.6 The calibration form

By clicking the “Save to Calibration File” button on the calibration form, the calibration data is saved to the file “uTracer_3pX.cal,” where “X” denotes the current GUI version number. The file is stored in the same folder in which the GUI executable is located, which is usually in the “program files” folder [\[MORE\]](#).

Tip:

Always record the calibration values on a piece of paper or in a text file! Although it is possible to copy & rename the old calibration file name to the new name after a new GUI release has been installed, it is by far quicker and easier to just re-enter the correct values after a new GUI release has been installed.

Note:

The COM port number is also stored in the calibration file. This means that if the COM port selection is changed, the calibration form must be opened and the “Save to Calibration File” button pressed if the new port selection is to be stored.

6.6 – Location of the Files

Windows XP:

The location where the calibration file is stored, and the default location where plots and data files are stored is in the same folder where the executable is stored. If the normal installation procedure is followed this is: "C:\Program Files\uTracer3px"

Windows 7: If the User Account Control (UAC) feature is enabled (which is default) then any attempt by an application to write to system directories is secretly re-directed to a user-specific "virtual store" that is read/write for that user. This is done to prevent users from corrupting installed applications, and the same thing has been done on UNIX for years.

The default location for your virtual store is:

"C:\Users\<>username>\AppData\Local\VirtualStore\Program Files" or

"C:\Users\<>username>\AppData\Local\VirtualStore\Program Files (x86)"

where <username> is your login username.

The virtual store is designed to shadow the directory structure under "C:\Program Files" but is qualified by the login username. If you are running 32-bit Windows 7 or have some 32-bit applications installed, you will see the "Program Files (x86)" directory, where 64-bit Windows 7 uses the "Program Files" directory.

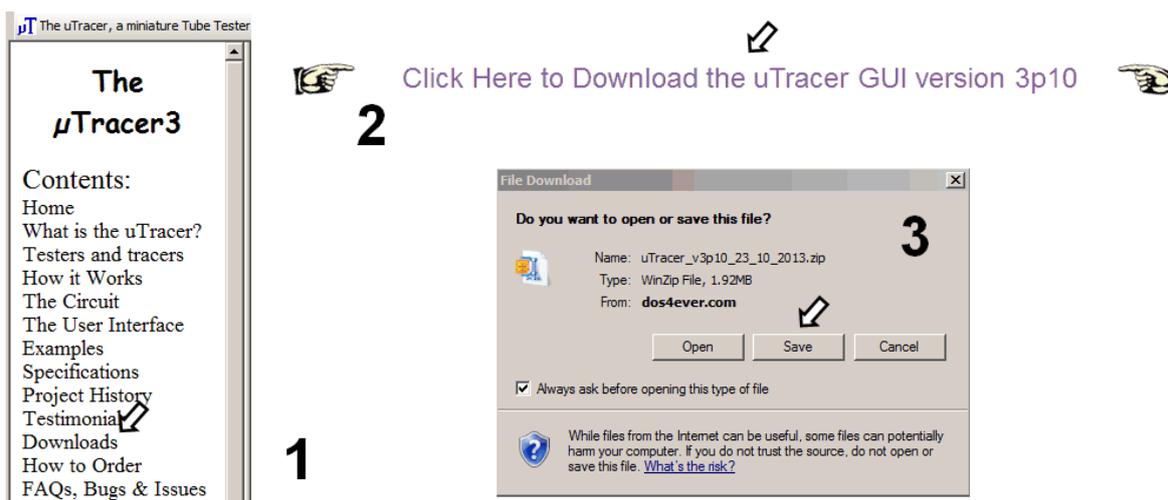
Note that the "AppData" directory is by default hidden from view, as it is a system folder. You can make it visible in Explorer by doing the following: Navigate to "C:\Users\" then from the "Organize" menu select "Folders and search options". From the dialogue box that then pops up, select the "View" tab. In the "Advanced setting" window under "Hidden files and folders", select the "Show hidden files, folders, and drives" radio button - then click "OK".

Under the "Program files" directory above, you will see a further directory called "uTracer_v3px". This is where the calibration file and plot/data files are stored.

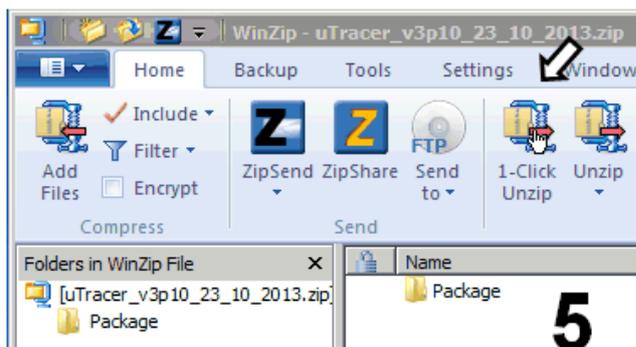
7 – Installation of the GUI

The installation of the GUI is normally quite straightforward. Just unzip the download to the desktop or a temporary directory. There are three files in the unzipped download: a CAB (cabinet) file, an LST listing file and a Setup application. The CAB file is just another zipped file that contains the GUI executable and some ActiveX components (.OCX files) that the executable needs to run. Most important are the mscomm32.ocx and MSCOMCTL.OCX, which deal with the serial communications. Normally double clicking the “SETUP” application should be sufficient to install the program and register the new components in the registry.

Here is the installation process for the GUI, illustrated with screen shots:



1. On the uTracer webpage navigate to the “Download” page.
2. Click the link to download the zip file with the GUI (The actual GUI version number may differ from the one displayed here).
3. Save the zip file to a suitable location, e.g. the desktop

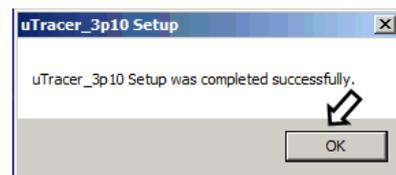
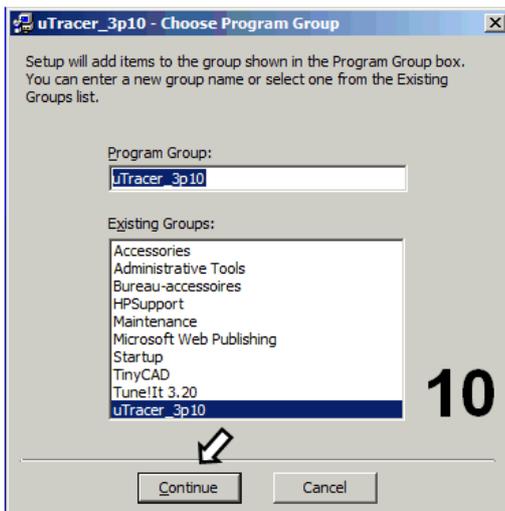
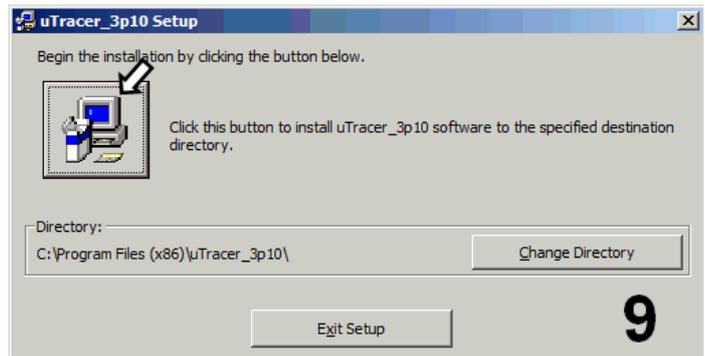


4. Double click the zip file to unzip the file e.g. using winzip.
5. Click “1-Click Unzip”, a folder with the setup files will be generated.
6. Open the folder.

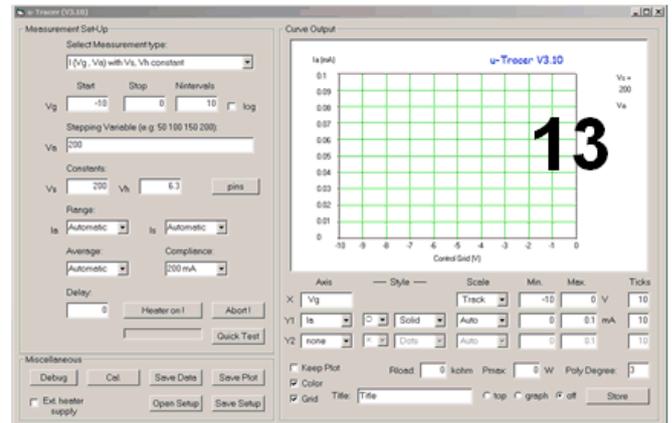
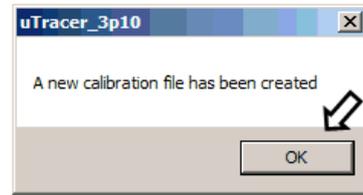
Name ^	Date modified	Type	Size
Package	3-11-2013 20:49	File folder	

Name ^	Date modified	Type	Size
setup	18-6-1998 1:00	Application	138 KB
SETUP	23-10-2013 20:46	LST File	4 KB
uTracer_3p10	23-10-2013 20:35	WinZip File	1,913 KB

- 7. Open folder Package
- 8. Double click the Setup application

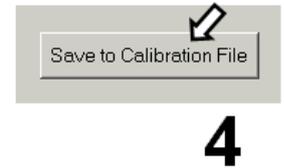
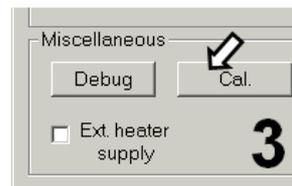
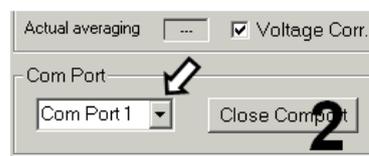
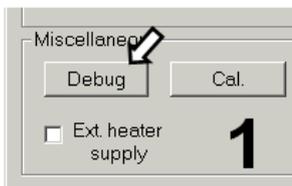


- 8,9,10. The installation program now installs the GUI. It basically only requires four OK clicks.
- 11. The GUI successfully installed!



- 12. A uTracer icon should have appeared in your Program list. Double Click it
- 13. On first time start, the GUI will create a blank calibration file. Click OK
- 14. The main form of the GUI should now appear.

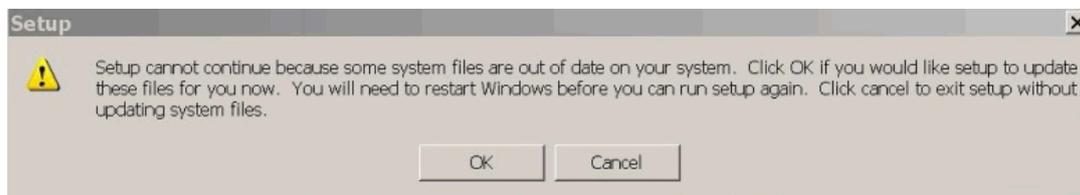
Lastly, the proper COM port has to be set so that the GUI can communicate with the uTracer. This is done on the communications form, which can be opened by pressing “Debug” in the main form (1). After selection of the correct COM port (2), the selected COM port number is saved in the calibration file by opening the calibration form (3) and clicking “Save to Calibration File” (4).



7.1 – Trouble shooting

Normally the installation of the GUI should proceed smoothly, but in practice some difficulties may be experienced. In this section some tips have been collected from users who found solutions to problems they experienced.

If the “Setup.exe” installer supplied with the download doesn’t work at all, you can create a directory “uTracer3px” in the root of your system by hand. This directory should be created in the “Program Files” directory for XP, and in “Program Files (x86)” For Windows 7. Then simply unzip the CAB file into this directory. With a bit of luck the necessary OCX components have been already installed and registered in the past by some other program, and the GUI will run by simply double clicking the executable “uTracer3px.exe”. Most of the time, however, one or two OCX components are missing and they will need to be installed by hand. Note that manual installation of these missing files is also required if during the normal setup procedure the following message is displayed:



Since this has never happened on one of my own systems, I can only relay the solutions others have found! The error message is probably related to a problem with your registry!

The first thing to do is check to see if the setup application has made a directory “uTracer3px” in your “Program Files” (Windows XP) or “Program Files (x86)” (Windows 7) directory. If not, you will have to create it yourself, and unzip the uTracer3px executable into it ([see previous section](#)).

Next the missing components need to be installed. Joe Neil has given me a description of how this can be done manually:

In order for this app to work, you need to "register" mscomm32.ocx with regsvr32.exe.

You should be able to do this with a command line but sometimes that doesn't work. Here's an alternate way:

For Win XP, regsvr32.exe should be in the \WINNT\system32 folder. mscomm32.ocx should be there as well, but you may need to download it.

I got it here http://www.nodevice.com/dll/MSCOMM32_OCX/item12152.html

To "register" mscomm32, put your cursor on the mscomm32 file, hold the left mouse button down, "drag" the file on top of the regsvr32 file, and release the button (drop). You should get a window that mscomm32 has been successfully registered.

7.2 – Installing a new release

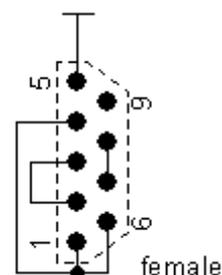
If you already have the GUI installed and working, there is no need to go through the complete installation procedure when a new version is released, since all the necessary OCX components have already been installed. It is enough to simply create a directory, preferably in the “Program Files” (Windows XP) or “Program Files (x86)” (Windows 7) directory, with a suitable name (e.g. uTracer_3p8), and unzip the “v3p8-executable-only-zip-file” (available on the [download page](#)) directly into that folder. Double click the executable and it should work.

I strongly recommend the following approach to port the calibration values to the new GUI: Start the old version of the GUI, and from the calibration form note down all the calibration values (on a piece of paper). Next, open the new version, set all of the calibration values accordingly, and press “Save to calibration file”. Exit and restart the GUI, then open the calibration form and verify that all the calibration values have been properly saved.

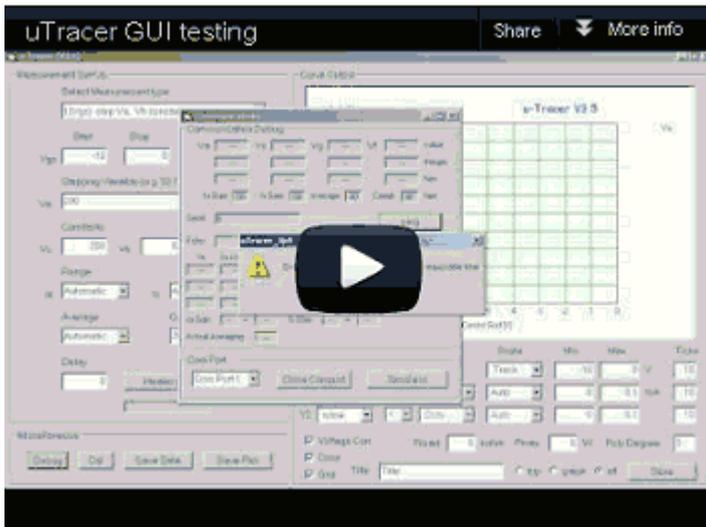
7.3 – Testing the USB to Serial converter

Unfortunately the use of a USB to serial converter has proven to be problematic. This is not the fault of the GUI or the uTracer, it is purely related to the converter hardware / software, sometimes in combination with the Windows installation. It is therefore recommended to first test the converter with the procedure described in this section.

A particular problem occurred with a USB to Serial converter that used the “Prolific” chipset. When the GUI tried to read the COM port, the program crashed with a “Run time error 8020, Error reading comm device.” A search on the Internet revealed that



this particular problem only occurs for USB to serial converters based on the “Prolific” chipset, and only under Windows Vista or Windows 7.



If you consider building your own uTracer I strongly recommend that you download the GUI and test it first. It is very easy to test the GUI and its communication without having the actual uTracer hardware by using an emulated COM port. Although it is possible to test the GUI with the COM port just open, a more thorough check is possible if the COM port is fooled by connecting the transmit pin to the receive pin. The small dummy uTracer circuit on the right shows how such a circuit can be wired on a 9-pin female RS232 (DB9) connector. Pins 2 and 3 are the data pins, and the other wiring makes sure that the handshake signals are properly bypassed.

After installing the GUI as described above, two configurations should be tried in order to have a 100% fail-safe test of the GUI communication:

With the COM port open, start the uTracer and click the “Debug” command button. Select the proper COM port number and press the “Ping” command button. Pressing the Ping button will cause the GUI to send two command strings to the uTracer. The first string is a “start measurement sequence” command, and the second string is a “read out all analog channels” command. The GUI will transmit the first character (the first “0” in “00”) and then wait for uTracer to echo it. When the COM port is open nothing is echoed, so after 2 seconds the GUI will detect a timeout and give a timeout error.

Connect the “dummy uTracer” circuit to the COM port, and press the “ping” command on the Debug form. In this case the GUI will successfully transmit the two command sequences, and then after 10 seconds give a timeout error because it expects a result string to be sent back by the uTracer.

If the GUI responds as above in both cases, the GUI and its communication with the COM port are functioning properly.

7.4 – Stacking of COM ports

It sometimes happens that some devices claim a COM port and then not “release” it when the device is removed. Especially (cheap) Bluetooth serial communications modules have this nasty habit. The result is that the lower COM ports numbers become occupied by devices which are not there. Since the COM port number selected on the GUI has to be in the range of 1-10, it can happen that you run out of COM ports. Here is a way to release unused COM port numbers:

1. First thing is to see if hidden devices are taking up the COM ports:
Click Start > All Programs > Accessories and then click Command Prompt.
2. Type “set devmgr_show_nonpresent_devices=1” in the Command Prompt and then press Enter.
(you might have to restart your computer after this)
3. Type “cd \Windows\system32” in the Command Prompt and then press Enter
4. Type “start devmgmt.msc” in the Command Prompt and then press Enter
5. Click View > Show hidden devices. This will display devices that are not connected to your computer
6. Now if you expand the section on COM ports, all the COM ports that have ever been created will be displayed, the non present ones being in grey.
You can uninstall away anything that you don’t want (right click, select uninstall).

8 – Capita Selecta

In this section a number of selected topics will be discussed which highlight some special features of the GUI.

8.1 – The communications form

10 = ok !

command string sent to – and echoed string returned from the uTracer

number of averages used

values sent to the uTracer

values sent back by the uTracer

gain code (02) and actual gain used (5x)

gain code (02) and actual gain used (5x)

Figure 8.1 Communication debug form.

The “Communications form” is opened by clicking “Debug” in the miscellaneous section on the main form. The top of the form displays detailed information about the communication between the GUI and the uTracer. This window was used extensively during the debugging phase of the program development, and I still use it regularly to check what is going on in detail. The form has three areas, the top part displays data that is sent to the uTracer by the GUI [\[MORE\]](#). In the four columns we find the set-points of the voltages which together determine the bias point of the tube. The top row gives the decimal value. The row below that gives an integer representation of the set-point which has been translated to a value which makes sense for the 10-bit on-chip AD/PWM converters. The row below that just gives the hex representation of the integer value. The last row gives the values for the gain, averaging and compliance settings which are sent to the uTracer in the command string. These last values are already coded so that the uTracer can easily interpret them.

The four set-points, headed by a command byte are combined into a command string which is shown in the middle section of the form [\[MORE\]](#). Normally the uTracer echoes every character sent to it. This process is monitored in the line below the command string which shows the echoed characters.

In the bottom section of the form, the data sent back by the uTracer in the result string is displayed. The result string is composed of three parts. It is headed by a status word which is displayed separately. A value of 10 indicates a successful measurement, while 11 indicates that a compliance error occurred. The bulk of the data in the result string is the readout of the 8 AD channels. The hex readout and the integer representations are shown in the top two lines. In the row beneath that the decimal values are shown, where the conversion from AD readout to real voltages and currents has been done. In the boxes at the bottom the actual gain and averaging values which were used for that particular measurement point are shown. They may vary from point to point when the automatic ranging and averaging options have been chosen, and it is instructive to see them change during a measurement.

In the bottom part of the form the COM port number can be selected from a drop down menu. Note that after a new COM port is selected, the number will not be saved to the calibration file unless the calibration form

is opened and “Save to Calibration File” is pressed. Furthermore, there is a button to “close” the current COM port or to send an escape character to the uTracer. An escape character always forces the uTracer to return to its reset state.

8.2 – Magic eyes & continuous measurements

The uTracer is actually very suitable for “testing,” a better word is “viewing,” magic eyes. To do this the uTracer is used in continuous mode. This has caused some confusion amongst users who were looking for a “continuous mode” button on the GUI. Here I will try to explain what is meant by it.

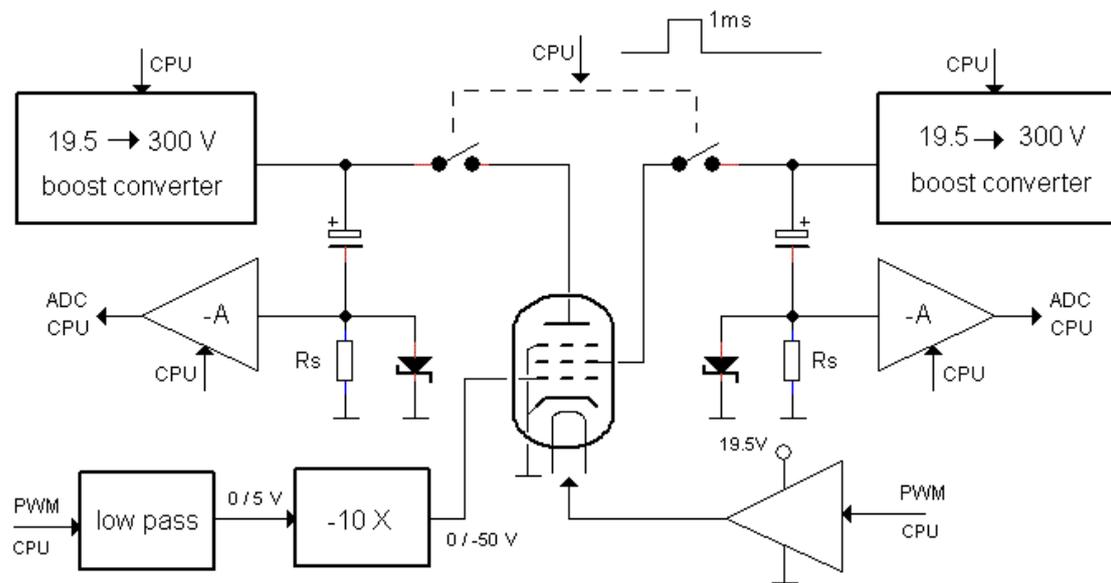


Figure 8.2 Normal pulsed operation of the uTracer

Figure 8.2 shows the normal “pulsed” operating principle of the uTracer. Two low-power boost converters charge two 100 uF reservoir capacitors to the desired anode and screen voltages. During the actual measurement the reservoir capacitors are connected to the anode and the screen by closing of high voltage switches for a duration of one millisecond. During this millisecond the anode and screen currents are supplied by the reservoir capacitors and the currents are measured.

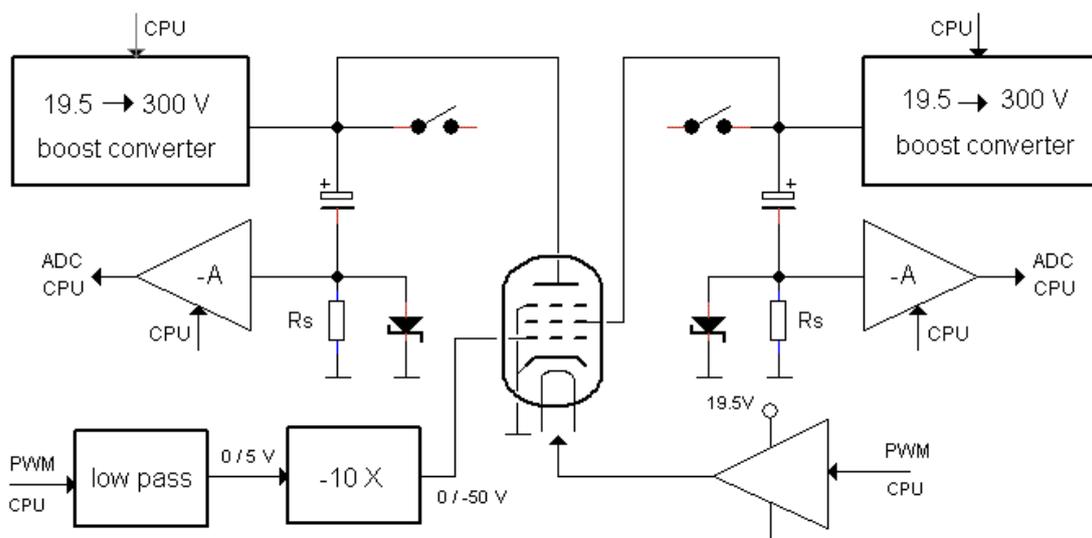


Figure 8.3 Continuous mode operation of the uTracer

In continuous mode operation (Fig. 8.3) the high voltage switches are not used, but instead the anode and the screen are directly connected to the boost converters. Since the boost converters can only supply

switching frequency is so high that the temperature of the filament cannot follow the switching, and in this way assumes a temperature which relates to the average dissipated power. By varying the duty-cycle, the amount of power dissipated in the heater can be varied. The relation between the duty-cycle, the set point heater voltage, and the supply voltage is derived [here](#). Note that the duty-cycle is equal to the square of the ratio of the set point voltage to the power supply voltage ([more here](#))! So, if a heater voltage of 10 V is specified, and the power supply voltage is 20 V, the duty-cycle is set to $(10/20)^2 = (0.5)^2 = 0.25$ or 25%!

Some people have tried to check the heater supply voltage of the uTracer with a normal voltmeter and find very strange values. That is because the output of the heater supply is pulsed. **You cannot check the heater supply voltage of the uTracer using an ordinary voltmeter!** Only a voltmeter which can measure the true rms (Root Mean Square) value of an AC signal with a sufficiently high bandwidth can be used. I don't have such a voltmeter, but a few people have reported on it ([Link1](#), [Link2](#)).

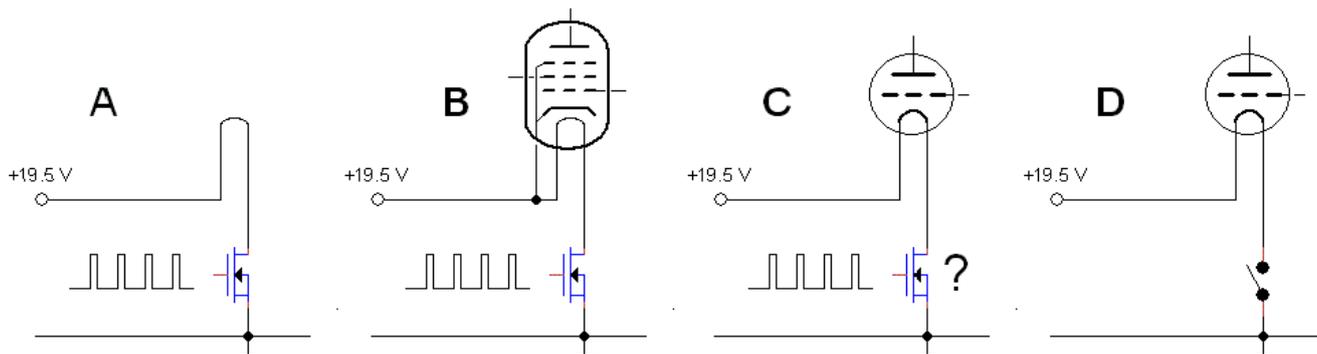


Figure 8.5 Principle of the heater supply

The heater supply of the uTracer obviously cannot be floating and has to be connected to the rest of the circuit. Figure 8.5 shows the principle of how that is done. Since the output voltage of a boost converter can never be lower than the supply voltage, the cathode of the tube is referenced to the positive supply voltage ([more here](#)). For indirectly heated tubes this is obviously no problem at all (Fig. 8.5B). For directly heated tubes there seems to be a problem because in this case the heater also serves as cathode, and what is the cathode potential if the heater is continuously switching on and off between zero and the maximum supply voltage (Fig. 8.5C)? The solution is simple: During the 1 ms measurement pulse when the high voltages are applied to the tube, the heater supply is completely switched off so the cathode/filament is at ground potential (Fig. 8.5D)! Note that this is a rather strange situation normally never occurring in any practical circuit using directly heated tubes. In normal circuits the heater voltage will always cause a voltage gradient along the filament resulting in a gradient of the grid-heater bias. So the currents in a real circuit will differ slightly from the currents measured with the uTracer! How serious that is depends on the heater voltage and the grid bias.

Low Voltages, High Currents.

A problem arises for heaters which combine a low voltage with a high current. The root of the problem is that as the heater voltage decreases, the duty-cycle of the PWM heater supply, and thus the heater supply pulse-width decreases according to an inverse square law. The pulse width becomes so short that even a small inductance in the wiring of the heater circuit prevents the heater current from rising to its maximum value. Some of this inductance may have even been added intentionally in the form of RFI suppression beads to avoid oscillations! Note that for higher heater voltages there is no problem at all. A 6.3 V / 1.5 A heater in an EL34 works perfectly, the problem is only in the combination of low-voltage and high-current. More information can be found in the [weblog section 23](#).

Several measures can be taken to minimize the problem:

1. For the heater connection do not use RFI suppression beads. Unfortunately this conflicts with an AVO style wiring of “tube-board” where RFI beads are included in every wire to suppress oscillations.
2. Use as short as possible and twisted wires for the connection between the uTracer and the tube socket.
3. Simply increase the set-point voltage for the heater! The question then is to what value. I will come back to that in a moment.
4. Use an external DC power supply. It is perfectly possible to do so, for both indirect as well as directly heated tubes. This is by far the best, and safest solution for both the tube, as well as the uTracer.

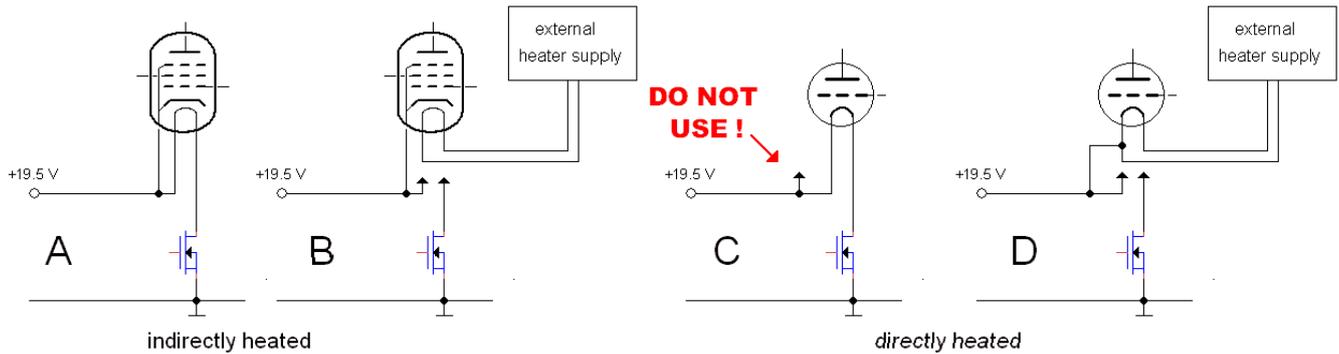


Figure 8.6 Connection diagram of indirectly / directly heated tubes with internal / external power supplies.

Testing Indirectly Heated Tubes

To test an indirectly heated tube with the uTracer, connect the heater and the cathode of the tube to the corresponding terminals of the uTracer (Fig. 8.6A). When an external heater supply is used, it can be simply connected to the heater of the tube, leaving the uTracer’s heater terminals unconnected. The cathode of the tube obviously remains connected to the uTracer (Fig. 8.6B).

Testing Directly Heated Tubes

To test a directly heated tube the heater of the tube is connected to the heater terminals of the uTracer, but the cathode terminal of the uTracer is not used (Fig. 8.6C)! Connecting the cathode terminal to one of the heater connections can short-circuit the heater supply, which in one case has destroyed a complete uTracer, **DON’T DO IT!**

The situation is different when an external heater supply which is electrically floating with respect to the uTracer is used. In this case the cathode connection must be connected to one of the heater connections on the tube (Fig. 8.6D). The resulting curves may vary depending on which heater terminal is connected to the cathode due to asymmetries in the tube. You may even consider placing a low-resistivity potentiometer in parallel with the heater, connecting the uTracer’s cathode terminal to the slider.

Another note of caution is needed when delicate battery tubes are measured! As explained before, the heater duty-cycle becomes very short for low heater voltages. Small variations in the duty-cycle will thus result in relatively large variations in the dissipated power. So when I measure delicate battery tubes like the DAF96 (1AH5), Russian pencil-tubes etc, I simply use a 1.5 V battery as heater supply, In that way nothing whatsoever can go wrong.

A Simple Calibration Procedure

A simple trick to compensate for the inductances in the circuit is just to increase the set-point value of the heater voltage until the proper heater temperature is reached! The question is how much the heater voltage has to be increased? Unfortunately that differs from tube to tube, and since everything so very much depends

on stray inductances, even from uTracer to uTracer. If it is your plan to test only a limited number of different types of tube, the following calibration procedure can be followed:

1. Connect a “known good tube” to the uTracer and use an external heater supply, set to the nominal heater voltage.
2. Measure a set of curves for relevant settings, and store the measurement.
3. Remove the external supply and connect the internal supply.
4. Iteratively increase the heater voltage until the curves overlap with the stored curves.
5. Note down the set-point heater voltage for this particular type of tube.

The external heater supply option.

In the previous section it was explained that the heater “problem” occurs for heater voltages which are small compared to the supply voltage (19.5 V). For low heater voltages this results in a very low duty-cycle of the PWM heater voltage which, in combination with inductances in the circuit and the high currents, causes the problems. Use of an external heater supply solves this problem, but also makes it impossible to do an automatic heater voltage sweep. Fortunately there is a kind of compromise solution. The trick is to use the external heater supply to feed the heater PWM circuit, and to set this supply voltage as low as possible. For instance, if a heater voltage sweep from 0 to 6 V is desired, the heater supply voltage should be set to something like 7 V. In this way the pulses of the PWM modulated heater voltage do not become too short.

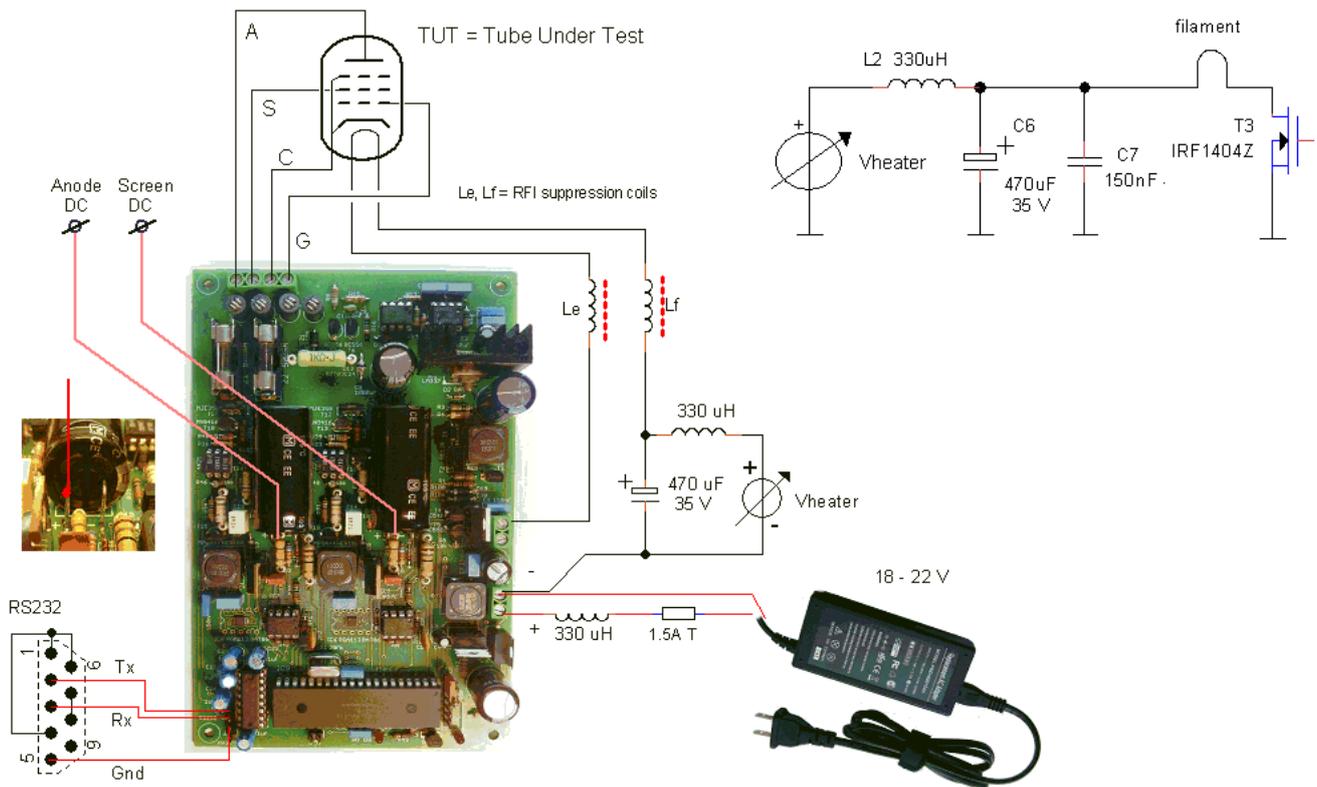


Figure 8.7 Connection diagram for the low-voltage external heater

Figure 8.7 explains the idea. The insert shows the heater circuit of the uTracer. Instead of using the 19.5 V supply voltage for the heater circuit, in this case an external power supply is used. The “floor plan” shows how with an extra inductor and a capacitor, the modification can be implemented without modifying the PCB. With a switch it is even possible to switch between the internal and the external heater supply. For the inductor any high current inductor of a few hundred uH can be used, e.g. one salvaged from an old PC power supply. *Note that this option only works in combination with tubes with an indirectly heated cathode since in this case the cathode remains connected to V_{supl} . For directly heated cathodes the heater is no longer referenced to V_{supl} , but to a lower potential.*

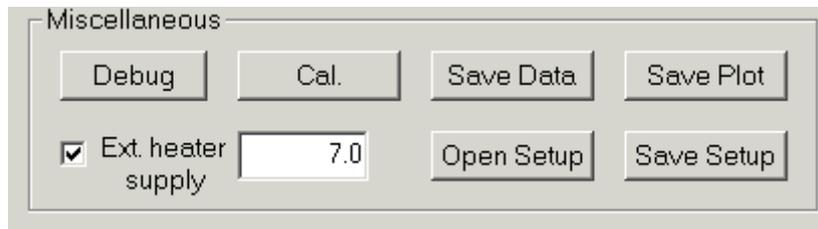


Figure 8.8 After activating the external heater supply option a field appears where the voltage of the external heater supply can be entered.

Using this method requires a special path in the GUI software because normally the 19.5 V supply voltage of the uTracer is taken as a reference. To activate this option tick the box “Ext. heater supl.” in the miscellaneous section on the main form. A new field will appear in which the voltage of the external heater supply can be specified. The actual heater voltage specified in the field “Vh” can now be set to any value between 0 and Vhsupl. If now under “Select Measurement type” a heater sweep is selected (I(Vh)), the heater voltage can be swept from 0 to maximal Vhsupl.

8.4 – The calibration procedure

Martin Manning has compiled the calibration processes for the uTracer into the following procedure (also available for download [here](#)):

PC-to-microcontroller Communication

If a calibration is to be performed, begin with this test:

1. Connect the uTracer to the PC
2. Connect the power supply to the uTracer (~20VDC, 1A)
3. Start the GUI
4. Switch on the uTracer power supply
5. Open the Debug/Communications window by pressing the “Debug” button in the Miscellaneous section of the main window
6. Select the appropriate Com port if necessary
7. Press the “Ping” button
8. Verify that the Debug form shows the following values: The “Send” and “Echo” strings should both read 500000000000000000 In the third row below “Vpower,” the supply voltage should appear. In the third row below “Vneg,” a value of approximately -40V should appear

Calibration Procedures

A calibration must be performed if the calibration data file is lost through a software re-load or upgrade, which will be indicated by all of the sliders on the calibration form being centered. Otherwise, a check of the calibration should be made on an annual basis.

Supply voltage

1. Measure the supply voltage using a DVM.
2. Compare the measurement to the voltage as displayed on the Debug form as in step 8) above.
3. If there is a discrepancy, open the calibration form by pressing “Cal” in the miscellaneous section of the main form.
4. Adjust the slider labeled “Vsupp” in the direction to reduce the error (i.e. move the slider to the right to increase the display value).
5. Press “Ping” on the Debug/Calibration form and observe the supply voltage display.

6. Repeat steps 4) and 5) as necessary until the display is in agreement with the DVM measurement.
7. Press the “Save to Calibration File” button on the calibration form.

Grid bias circuit

Warning: High voltages will be present in the uTracer circuit during this test!

1. Connect a DVM to the uTracer’s cathode terminal (positive lead) and grid terminal (negative lead).
2. Select measurement type I(V_g , V_a), with V_s , V_h constant, and set measurement parameters for V_g Start = -40, Stop = 0, Nintervals = 1, V_a = 200, V_s = 200, V_h = 6.3. Set Range and Averaging = Auto, Compliance = 200 mA, and Delay = 5 (sec).
3. Open the Calibration form by pressing the “Cal” button in the miscellaneous section of the main form.
4. Switch on the uTracer power supply.
5. Press the “Heater On!” button on the main form twice. The second press of the “Heater On!” button will cause the uTracer to skip the delayed heating function, which is not necessary for this test.
6. Press “Measure Curve” (which was formerly “Heater On!”) on the main form.
7. Read the actual grid voltage on the DVM during the -40V phase of the measurement.
8. Adjust the V_{grid} gain slider on the calibration form in the appropriate direction to close any discrepancy between the DVM measurement and the -40V set value (i.e. move the slider to the right to increase the measured grid voltage).
9. Repeat steps 6) through 8) as necessary until the measured voltage is -40V.
10. Press the “Save to Calibration File” button on the calibration form.

The set-up for the grid bias voltage calibration can be saved under an appropriate name such as “Cal Grid Bias” for future use.

Screen grid boost converter

Warning: High voltages will be present and measured during this test!

1. Connect a DVM to the exposed leads of the screen reservoir capacitor C13.
2. Switch the uTracer power supply on and note the idle state voltage on the DVM.
3. Select measurement type I($V_a=V_s$, V_g), with V_h constant, and set measurement parameters for $V_a = V_s$ Start = 100, Stop = 200, Nintervals = 1, $V_g = -1$, $V_h = 6.3$. Set Range and Averaging = Auto, Compliance = 200 mA, and Delay = 5 (sec).
4. Open the Calibration form by pressing the “Cal” button in the miscellaneous section of the main form.
5. Press “Heater On!” twice.
6. Press “Measure Curve” and observe the voltage measured by the DVM during the 200V phase of the measurement, comparing it to a value of $V_{idle} + 200V$, where V_{idle} is the voltage observed in step 2).
7. Adjust the V_s Gain slider on the calibration form in the appropriate direction to close any discrepancy between the DVM measurement and $V_{idle} + 200$ (i.e. move the slider to the right to increase the measured screen voltage).
8. Repeat steps 6) and 7) as necessary until the voltage measured during the 200V phase equals $V_{idle} + 200V$.
9. Press the “Save to Calibration File” button on the calibration form.
10. After the “HV On” indicator goes out, switch off the uTracer power supply and disconnect the DVM.

Anode boost converter

Warning: High voltages will be present and measured during this test!

1. Repeat the procedure above, except in step 1) connect the DVM to the anode reservoir capacitor C18, and in step 7) adjust the Va Gain slider on the calibration form so that the DVM measurement equals $V_{idle} + 200V$.

The set-up for the screen and anode voltage calibrations can be saved under an appropriate name such as “Cal Boost Converters” for future use.

Current amplifiers

Warning: High voltages will be present during this test!

1. Make sure that the power supply is off and that the reservoir capacitors are discharged
2. Connect a 10k, 1% resistor between the anode terminal and the cathode terminal
3. Connect a 10k, 1% resistor between the screen terminal and the cathode terminal
4. Select Measurement type $I(V_a = V_s, V_g)$, with V_h constant, and set measurement parameters for $V_a = V_s$ Start = 195, Stop = 210, Nintervals = 4, $V_{gs} = -1$, $V_h = 6.3$. Set Range $I_a = 0 - 40$ mA, $I_s = 0 - 40$ mA, Average = 4X, Compliance = 200 mA, and Delay = 0 (sec).
5. Set plot controls: Display I_a on the left Y-axis and I_s on right Y-axis. Set all axes scale ranges to manual. Set the X-axis scale range for 190 to 210V, with 2 tick marks. Set both Y-axis scale ranges for 19 to 21 mA, with 2 tick marks.
6. Open the calibration form by pressing “Cal” in the miscellaneous section of the main form.
7. Switch on the uTracer power supply and press “Heater On!” twice.
8. Press “Measure Curve” on the main form to start the measurement.
9. Observe the resulting plot and adjust the I_a and I_s gain sliders on the calibration form in the appropriate direction so that the straight lines on the plot (indicating a pure resistance) will pass through 200V at 20 mA at the center of the plot.
10. Repeat from step 8) until both the I_a and I_s lines pass through 200V and 20 mA.
11. Press the “Save to Calibration File” button on the calibration form.

The set-up for the current amplifier calibrations can be saved under an appropriate name such as “Cal Current Amps” for future use.

8.5 – Positive grid bias (A2 mode)

This section on positive grid biasing is in preparation.

For the moment please refer to the [section on positive grid biasing](#) on the weblog page.

(scroll to section 25 if your browser has difficulties finding the right entry point in long pages)

8.6 – Suppressing oscillations

This section on suppressing oscillations is in preparation.

For the moment please refer to the comments in the construction manual which can be downloaded [here](#).

Special thanks to Martin Manning for proof-reading and correcting this manual!