Rohde & Schwarz UPL-B10
Sequence Controller

Table of Contents

1. Intro

The UPL-B10 enables the UPL analyzer to run BASIC programs, log a sequence or run macros. It makes automatic setups accessible without human interaction or external GPIB/Serial communication.

Since manual inputs will be key, an external keyboard will be required. The UPL itself only allows for very minor adjustments but generally no program edits.

2. R&S BASIC

Throughout the manual we'll mention the UPL's BASIC environment called "R&S BASIC".

Important to note is that this is an embedded version of BASIC. The BAS files made on the UPL can only be read or modified on the UPL itself. Not all existing BASIC commands are supported but enough to make simple programs. There is also a command difference with the IEC commands. I recommend to always check the manual or use the function logging described below.

3. Function Logging

To start programming in BASIC you'll need some understanding of C but most important of all you'll need to know the many functions to address the UPL's potential. You can explore the 800-page manual or you can use the function logging and let the UPL help you.

3.1 Start Logging

To start the logging press the F2 key on the external keyboard. On the right bottom of the screen the UPL will show "Logging on".

From this point you can set up your analyzer and all steps will be recorded.

Important to note is that when you load a SAC (setup) file during this step, all actions from the SAC file will be recorded. This might make your logging very cluttered with items you are probably not interested in.

3.2 Stop Logging

To stop the logging press the F2 key again. This will be shown on the right bottom of the screen as "Logging off".

3.3 Show Logging

To show the different steps you need to go to the R&S Basic Environment by pressing F3 or Backspace on the keypad of the UPL.

The recorded steps will be shown on the screen.

4. Macro

To run compiled BASIC programs easily you can use the MACRO command. This can be found under the OPTIONS menu "Exec Macro". When using this there is no need to go to the R&S BASIC environment and loading the program by typing the name. This step can be very useful when you want to execute your program without having a keyboard connected.

Important to note is that only BAS files can be loaded which are made within the R&S BASIC environment.

5. Programming

In order to start programming I have listed the flow which can be followed.

5.1 Write your program

Inside the R&S BASIC environment you have the option to save your recorded steps as a TXT file which can be modified on any editor.

5.2 Run your program - RUN

You can always run your program by pressing F6 or using the Function keys underneath the display. Alternatively, you can type RUN underneath your program and push enter.

5.3 List your program - LIST

The display is limited so lines will generally not be displayed if you have more lines than the screen can display at once.

After loading or running a program you can always reload what is in memory by pressing F8 to LIST the program.

5.4 Clear the screen - CLEAR

If too much text is on the screen and you want to start from scratch, you can type CLEAR and press enter. This will clear out the entire screen. By Listing your program again, you can start over.

5.5 Save as TXT File - ASAVE

When you have finished your program, you can save it as a TXT file. This is important as it serves a backup which can be modified on a normal PC.

To save as a TXT File you'll have to use the "ASAVE" command. Note that the A stands for ASCII save which means TXT file.

e.g. ASAVE"A:Myprogram.txt"

5.6 Load as TXT File – ALOAD

When you want to run a TXT file you can do so by using the ALOAD command. Note that the A stands for ASCII load which means TXT file.

After that you can LIST the program or RUN it.

e.g. ALOAD"A:Myprogram.txt"

5.7 Save as BASIC File – SAVE

When you want to save a BASIC file, you'll need to use the SAVE command. Note that there is no A. This means save as a BAS file.

e.g. SAVE"A:Myprogram.bas"

5.8 Load as BASIC File – LOAD

When you want to load a BASIC file for running or modification you can do so by using the LOAD command. Note that there is no A. This means load as a BAS file.

e.g. LOAD"A:Myprogram.bas"

6. Convert TXT file to BAS file

If you want to use your program in the MACRO option for example, you'll need a BAS version of your program.

This can be simply done by loading your TXT file program in R&S BASIC by using ALOAD. If you then use the SAVE command. Note that there is no A. You'll save a version of your program as a BASIC file.

e.g. SAVE"Myprogram.bas"

You'll have a functional BASIC version of your program which can be used as a MACRO.

7. Other Important BASIC programs

There are several useful BASIC programs that can be found.

7.1 SELFTEST.BAS

During the startup of the UPL analyzer a short self-test is executed. When troubleshooting the device, a more extended self-test is preferred.

This can be found when executing the MACRO and under the C:\UPL\USER path you can find the SELFTEST.BAS file.

7.2 FLAT_GEN.BAS

When first receiving the unit there will be a compensation curve available for the generator. This is part of the calibration routine at R&S.

When having the UPL-B10 option you can re-run this program at any time. There are no cables attached as the routing is done internally through the relays.

8. Tips

Here are some tips I've gathered. Some may be obvious others can be useful.

- To stop a program, you can use the CTRL-C combination. If the UPL is busy with a command it will finish executing it and at the next available step the program will be halted.
- String Variables are noted as MyString$.
- Integer Variables are noted as MyInt%
- Double Variables require no special notation. Assigning a value to it finishes the declaration. E.g. Voltage=0.00
- Always remember the differences between ASAVE and SAVE and ALOAD and LOAD.
- Always wait to remove floppy or USB drive until all activity is done. Not doing this has ended up several times with corrupted files.
- Use the R&S BASIC environment to start your program, record steps. Use notepad++ on your PC for everything else.
- Avoid using TABS in your files. Use lots of spaces if needed.
- Load a SAC File to do 90+ % or your work. Use your program to do small things. Loop, Trigger measurements, Logging, Change key parameters (Voltage, Frequency, etc.). Be as lazy as you can.

9. Examples

9.1 Example 1

This example will loop a finite number of times a measurement. During each loop the voltage will be increased, the measurement triggered, a unique filename generated and the Traces saved by the filename generated before.

This example shows good use of variables for key parameters e.g. Voltage as well as generic tasks as generating an increasing filename that will be use later on in the loop. Also note the PRINT actions serving as debugging steps.

```
10 '   *****************************************************************
20 '   * BVKSound 2020/05/09                                 *
30 '   * Example for trigger multiple meas. and save trace files       *
40 '   * 2-channel trace file will be saved on A-Drive              *
50 '   * Voltage increases each iteration by 10mV                   *
60 '   *****************************************************************
70 '
80 '
90 Trcprefix$="A:Output"
100 UPL OUT "MMEM:LOAD:STAT 0,'A:MySetup.SAC'"
110 '
120 '
130 Volt=0.01
140 '
150 FOR I=1 TO 10
160  Trc$=Trcprefix$+STR$(I)+".TRC"
170  PRINT "Measurement" + STR$(I)+" \ Voltage:"+ STR$(Volt) + "V \ FileName: " +Trc$
180  UPL OUT "SOUR:VOLT "+STR$(Volt)+""
190  UPL OUT "INIT:CONT OFF;*WAI": 'Triggers single measurement
200  UPL OUT "MMEM:STOR:TRAC TR1A,'"+Trc$+"'"
210  '
220  Volt=Volt+0.01
230 NEXT
240 '
250 '
260 PRINT "Program Finished"
270 END
```