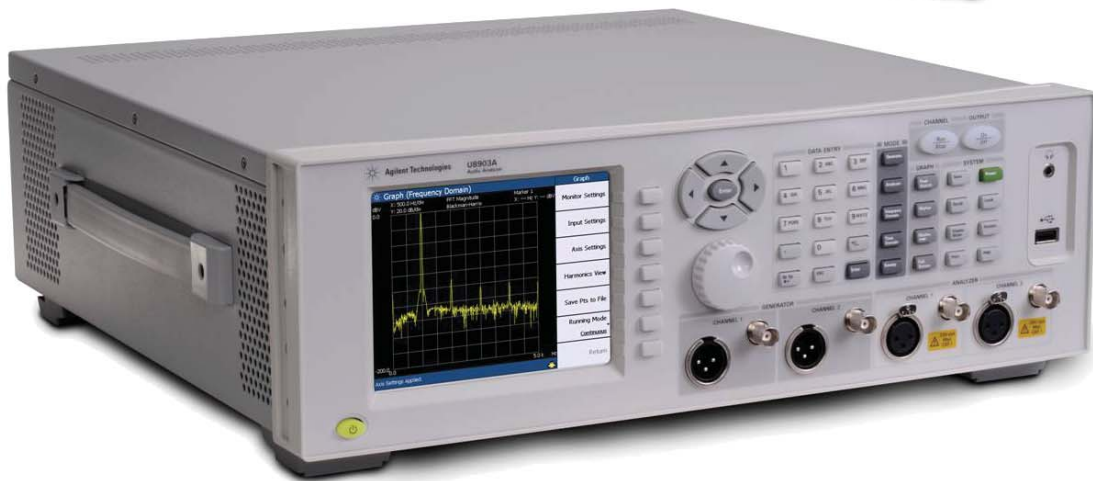




# Migrating Code from the 8903B to the U8903A

## Application Note



### Table of Contents

Introduction.....	2
Fundamental Differences .....	2
Initializing the Instrument .....	3
Setting Source Frequency and Amplitude.....	3
Programming a Typical Measurement.....	3
Setting Measurement Filters .....	4
Supporting Service Requests.....	5
Programming Example .....	7
Unsupported Commands .....	9
Hints and Tips.....	9
Conclusion.....	10
Appendix.....	11

The Agilent U8903A audio analyzer, introduced in 2009, is the successor to the 8903B. The U8903A implements the modern SCPI standard for its command set, which is considerably different from the instrument-specific command set used by the 8903B. This application note will assist you in migrating your 8903B code to work with the U8903A. It highlights the basic differences between the two command sets, and explains how to convert your legacy code for basic programming tasks. A full programming example for a typical audio analyzer application is also provided, along with hints and tips for a successful migration.



**Agilent Technologies**

## Introduction

When Hewlett-Packard introduced the HP 8903B audio analyzer more than two decades ago, it quickly became the industry standard because of its unprecedented performance and features for its time. Its enduring popularity, even after Agilent Technologies inherited HP's measurement business in 1999, made compatibility with the 8903B a high priority in the design of Agilent's next generation audio analyzer. The Agilent U8903A is not only designed as a replacement for the 8903B, it also includes up-to-date features such as FFT analysis, making it ideally suited for the demands of modern audio test.

To ensure that migrating from the 8903B to the U8903A is as effortless as possible, Agilent has taken steps to ensure physical and features compatibility. For example, the U8903A not only fits into the same 3U rack space, it also has the same power-up conditions as its predecessor.

Command set compatibility is another important consideration when replacing an instrument, especially if it is used in an automated test system. The limited processing power available at the time compelled the designers of the 8903B to employ a simple, instrument-specific command set for remote programming. These commands are difficult to learn and use. Therefore, it was decided not to reuse the existing command set for the U8903A, but to use Standard Commands for Programmable Instruments (SCPI) instead.

SCPI emerged as the standard for instrument programming in the early 1990s. Following a hierarchical structure with subsystems, SCPI groups associated commands together under a common node or root. Its keywords are common English words and not cryptic acronyms, making SCPI programs easy to understand and code.

## Fundamental Differences

The 8903B command set has its own unique syntax, where command keywords are concatenated with parameter values and units, without any delimiters such as colons or spaces. The commands can be written in several ways:

### Special function code

`<numeric_special-function_code>SP`

Example: `1.0SP`                    *Sets range to auto*

### Standalone command

`<command>`

Example: `M2`                        *Sets SINAD measurement*

### Command with parameter

`<command><numeric_value><unit>`

Example: `FR1KZ`                    *Sets output frequency*

Two or more commands can be concatenated in a single line without any separators. For example, the command string below is perfectly valid.

```
1.0SPM2FR1KZ
```

SCPI commands, on the other hand, conform to a set of rules. A command is delimited from its parameters by a space. A colon (:) separates a command keyword from its lower level keyword. Commas are used as delimiters for two or more parameters. In SCPI, multiple commands can also be written in the same statement, just as for the 8903B, except that a semicolon (;) must be used to separate one command from the other.

The conventional syntax used when describing SCPI commands is as follows. Most commands (and some parameters) will be shown as a mixture of uppercase and lowercase letters. The uppercase letters indicate the abbreviated spelling for the command. The SCPI parser accepts both abbreviated and long forms. Angle brackets ( < > ) indicate that you must specify a value for the enclosed parameter.

Some parameters are enclosed in square brackets ( [ ] ). These brackets indicate that the parameter is optional and can be omitted. A vertical bar ( | ) separates multiple parameter choices. As the U8903A is a multi-channel instrument, a channel list, in the form (@<channel\_number>) needs to be appended to most commands to indicate the channel.

As an example, the syntax for setting the channel 1 source frequency is:

```
SOURce:FREQuency <freq>[<unit>], (@1)
```

where <freq> is a numeric value from 5 to 80 000, and <unit> can either be HZ or KHZ.

The following statements are all valid examples of setting the source frequency to 1 kHz, which are the equivalent of the 8903B `FR1KZ` command.

### Long form

```
SOURCE:FREQUENCY 1KHZ, (@1)
```

### Abbreviated form

```
SOUR:FREQ 1KHZ, (@1)
```

### Unit omitted, default is Hz

```
SOUR:FREQ 1000, (@1)
```

Although any of the two channels on the U8903A can be used to replace the single channel on the 8903B; for consistency, channel 1 will be used in all the programming examples.

## Initializing the Instrument

The beginning of a test program would typically contain code to initialize all instruments. In the case of the 8903B, the IEEE-488 (GPIB) Clear interface function is used at the beginning to reset the instrument to its default state.

```
CLEAR 7
```

The use of this Clear interface function does not conform to the IEEE-488.2 standard. The U8903A complies with the standard by using the \*CLS (clear status) and \*RST (reset) common commands to perform the same task.

An additional command, `OUTP:STAT ON, (@1)`, is also required to turn the output on. This is necessary because for safety reasons, the outputs of the U8903A are disabled at power up (note that the 8903B does not have the capability to turn its output off). Therefore, to maintain compatibility, the start of the U8903A test program should always have the following commands:

```
ASSIGN @Aud to 728
OUTPUT @Aud; "*CLS; *RST"
OUTPUT @Aud; "OUTP:STAT ON, (@1) "
```

## Setting Source Frequency and Amplitude

After initialization, the next task is to set up the source to provide a stimulus to the device under test. The U8903A source channels can generate a multitude of waveforms such as sine, square, dual sine, and arbitrary. However, since the 8903B can only output a sine wave, and since this is already the default setting for the U8903A, you need not explicitly send a command to set the waveform type in SCPI. You would typically only need to send commands to set source frequency and amplitude, which are relatively straightforward. However, you need to take note of several constraints. As the U8903A targets audio-specific applications, it prioritizes dynamic range over frequency range. Due to this, the U8903A source frequency is limited to 80 kHz. If you attempt to set the source frequency to any value above 80 kHz, it will be clamped to 80 kHz.

The 8903B amplitude unit can be set to DV, which is equivalent to “dBm into 600 ohm”. The U8903A has no equivalent unit — the closest being dBV, which is “voltage relative to 1 V”.

To convert DV to dBV, use the following formula:

$$\text{dBV} = \text{DV} - 2.218 \text{ dB}$$

For example, if the 8903B has the following command to set its source amplitude to 0 dBm:

```
AP0DV
```

then the equivalent U8903A command is:

```
SOUR:VOLT -2.218DBV, (@1)
```

Table 1 lists the 8903B source unit parameters and their SCPI equivalent.

Table 1. Equivalent source units

8903B unit	U8903A unit
<b>Frequency</b>	
HZ	HZ
KZ	KHZ
<b>Amplitude</b>	
VL	VRMS
MV	MVRMS
DV	DBV

## Programming a Typical Measurement

After setting up the source, the next task is to program the audio analyzer to make a measurement. This typically consists of the steps as shown in Figure 1.

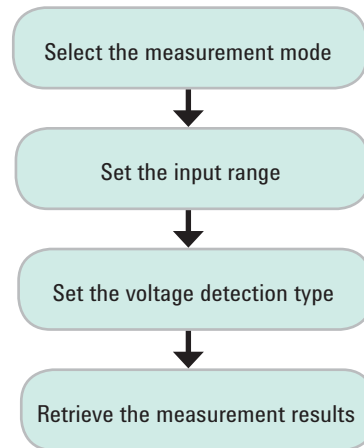


Figure 1. Typical measurement flow

Table 2. Typical measurement program sequence

8903B program sequence	U8903A program sequence	Note
OUTPUT @Aud;"M1"	OUTPUT @Aud;"SENS:FUNC2 VAC, (@1) "	[1]
OUTPUT @Aud;"1.11SP"	OUTPUT @Aud;"SENS:VOLT:RANG 3.2V, (@1) "	[2]
OUTPUT @Aud;"5.7SP"	OUTPUT @Aud;"SENS:VOLT:DET QPK, (@1) "	[3]
WAIT 1	WAIT 1	[4]
	OUTPUT @Aud;"INIT:ANAL (@1);:FETC? FUNC2, (@1) "	[5]
ENTER @Aud;Volt	ENTER @Aud;Volt	[6]

#### Notes

1. Sets AC voltage measurement
2. Sets 3 Vrms input range for the 8903B or 3.2 Vrms input range for the U8903A
3. Sets quasi-peak detector type
4. Waits for signal to settle
5. Initiates measurement, then transfers reading to the output buffer (U8903A only)
6. Retrieves measurement

Table 2 shows the equivalent 8903B and U8903A program sequences based on the flowchart shown in Figure 1. As you can see, migrating the code is not too difficult, as long as you are aware of several key differences. Firstly, the U8903A maximum input range is 140 Vrms. If your existing application requires you to make measurements above 140 Vrms, you would need to use an external attenuator with the U8903A. The second statement of the 8903B program sets the input range to 3 Vrms, which indicates that the measured signals are below 3 Vrms. This means that the U8903A is able to replace the 8903B directly without an external attenuator.

The corresponding U8903A statement sets the input voltage range to 3.2 Vrms, and not 3 Vrms. You cannot set the same value because the input ranges for both instruments are different. Setting the U8903A one range higher than the 8903B will ensure no overrange occurs. However, if you have knowledge of the exact voltage levels being measured, then you can modify the code to set the U8903A to a lower range. This will ensure a wider dynamic range for your measurement.

Finally, test system developers used to programming modern instruments might be a little puzzled by the last statement in the 8903B program sequence. The 8903B, in its default free run trigger mode, is able to read the measurement result directly from the bus without having to send a query command first. This runs contrary to modern instrument programming convention, as established by the SCPI standard. With the U8903A, you need to INITiate the measurement, and then query using the FETCh? command before retrieving any data from the bus. The equivalent U8903A program therefore has one extra line of code.

## Setting Measurement Filters

The 8903B allows up to two filters to be selected at a time: a built-in low pass filter and a plug-in filter. The low pass filter can either be a 30 kHz or 80 kHz low pass, while there are two plug-in slots that allow any one of five different weighting filters, or a 400 Hz high pass filter to be installed. Only one plug-in filter can be selected at a time.

If the 30 kHz low pass filter is selected, migrating the code is straightforward. The 8903B command:

```
OUTPUT @Aud;"L1"
```

is converted to the U8903A command:

```
OUTPUT @Aud;"SENS:FILT:LPAS LP30, (@1) "
```

To turn off this filter, the equivalent of the U8903A command, OUTPUT @Aud;"LO" is:

```
OUTPUT @Aud;"SENS:FILT:LPAS NONE, (@1) "
```

Selecting the 80 kHz low pass filter though, is a two-step process on the U8903A. As the 80 kHz filter is not built into the U8903A, you need to first define it as a custom filter. After that, you invoke the custom filter using the following command:

```
OUTPUT @Aud;"SENS:FILT:LPAS CUST, (@1) "
```

As for the plug-in filters, you first need to determine which of the six available filters are installed in the two slots. For example, if the A-weighting filter is installed in the left slot (H1), while the CCITT filter is in the right slot (H2), then the 8903B to U8903A code conversion is straightforward, as shown in Table 3.

Table 3. Code migration for weighting filters

8903B program sequence	U8903A program sequence	Note
OUTPUT @Aud;"M2"	OUTPUT @Aud;"SENS:FUNC2 SINAD, (@1) "	[1]
OUTPUT @Aud;"H1"	OUTPUT @Aud;"SENS:FILT:WEIG AWE, (@1) "	[2]
WAIT 1	WAIT 1	[3]
OUTPUT @Aud;"T2"	OUTPUT @Aud;"INIT:ANAL (@1);:FETC? FUNC2, (@1) "	[4]
ENTER @Aud;Sinad1	ENTER @Aud;Sinad1	[5]
OUTPUT @Aud;"H2"	OUTPUT @Aud;"SENS:FILT:WEIG CCIT, (@1) "	[6]
WAIT 1	WAIT 1	[7]
OUTPUT @Aud;"T2"	OUTPUT @Aud;"INIT:ANAL (@1);:FETC? FUNC2, (@1) "	[8]
ENTER @Aud;Sinad2	ENTER @Aud;Sinad2	[9]

Notes

- |  |   |
|--|---|
| <ol style="list-style-type: none"> <li>1. Sets SINAD measurement</li> <li>2. Applies A-weighting filter</li> <li>3. Waits for signal to settle</li> <li>4. Triggers measurement</li> <li>5. Retrieves measurement</li> </ol> | <ol style="list-style-type: none"> <li>6. Applies CCITT filter</li> <li>7. Waits for signal to settle</li> <li>8. Triggers measurement</li> <li>9. Retrieves measurement</li> </ol> |
|--|---|

The 8903B program sequence in Table 3 uses the T2 (trigger immediate) command to trigger a new measurement. The U8903A equivalent commands are INITiate and FETCh?, as they are used for the example in Table 2.

The program conversion becomes a bit more complicated if you have a weighting and high pass filter combination in the plug-in slots. This is because on the 8903B, selecting the H2 filter will automatically turn off the H1 filter, as only one plug-in filter can be selected at a time. This does not pose a problem if both slots contain weighting filters, because the U8903A will also only allow one weighting filter to be selected at a time. However, migrating the code is not so straightforward if one of the 8903B plug-in filters is the 400 Hz high pass, while the other is a weighting filter. This is because the U8903A allows a high pass and a weighting filter to be combined. To maintain compatibility, you need to turn off one of the filters on the U8903A before selecting the other one.

To illustrate this better, if the left plug-in slot contains the 400 Hz high pass filter, while the right slot contains the CCITT weighting filter, then the code conversion must be performed as shown in Table 4.

## Supporting Service Requests

The U8903A, just like the 8903B, supports GPIB service requests. However, unlike the 8903B, which has only one status byte (as shown in Table 5), the U8903A has five registers in its status reporting system. Besides the status byte (\*SRE), the U8903A also uses the standard event status register (\*ESE) for reporting the same conditions as the 8903B. Table 6 shows the equivalent commands for setting the service request conditions.

To enable more than one condition, you must sum the relevant bits. For example, to request for service if a command or instrument error occurs, you will program the following 8903B code:

```
ENABLE INTR 7;2
ON INTR 7 GOSUB Errorsub
OUTPUT @Aud;"22.6SP"
```

Sums bits 1 & 2

The equivalent code for the U8903A is as follows:

```
ENABLE INTR 7;2
ON INTR 7 GOSUB Errorsub
OUTPUT @Aud;"*ESE 32;*SRE 36"
```

Sums bits 2 & 5

Table 4. Code migration for a high pass and a weighting filter

8903B program sequence	U8903A program sequence	Note
OUTPUT @Aud;"M2"	OUTPUT @Aud;"SENS:FUNC2 SINAD, (@1) "	[1]
OUTPUT @Aud;"H1"	OUTPUT @Aud;"SENS:FILT:HPAS HP400, (@1) "	[2]
WAIT 1	WAIT 1	[3]
OUTPUT @Aud;"T2"	OUTPUT @Aud;"INIT:ANAL (@1);:FETC? FUNC2, (@1) "	[4]
ENTER @Aud;Sinad1	ENTER @Aud;Sinad1	[5]
	OUTPUT @Aud;"SENS:FILT:HPAS NONE, (@1) "	[6]
OUTPUT @Aud;"H2"	OUTPUT @Aud;"SENS:FILT:WEIG CCIT, (@1) "	[7]
WAIT 1	WAIT 1	[8]
OUTPUT @Aud;"T2"	OUTPUT @Aud;"INIT:ANAL (@1);:FETC? FUNC2, (@1) "	[9]
ENTER @Aud;Sinad2	ENTER @Aud;Sinad2	[10]

Notes

- |                               |   |
|-------------------------------|---|
| 1. Sets SINAD measurement     | 6. Turns off high pass filter (U8903A only) |
| 2. Applies high pass filter   | 7. Applies CCITT filter                     |
| 3. Waits for signal to settle | 8. Waits for signal to settle               |
| 4. Triggers measurement       | 9. Triggers measurement                     |
| 5. Retrieves measurement      | 10. Retrieves measurement                   |

Table 5. 8903B status byte

Bit	7	6	5	4	3	2	1	0
Condition	<N/A>	Require Service	<N/A>	<N/A>	<N/A>	Instrument Error	HP-IB Code Error	Data Ready

Table 6. Equivalent commands for setting service request conditions

8903B command	U8903A command	Description
22.1SP	*SRE 16	Requests service when data is ready <b>8903B:</b> Bit-0 of the status byte <b>U8903A:</b> Bit-4 of the status byte
22.2SP	*ESE 32; *SRE 32	Requests service when a command error occurs <b>8903B:</b> Bit-1 of the status byte <b>U8903A:</b> Bit-5 of the status byte and bit-5 of the standard event status register
22.4SP	*SRE 4	Requests service when an instrument error occurs <b>8903B:</b> Bit-2 of the status byte <b>U8903A:</b> Bit-2 of the status byte

## Programming Example

This section demonstrates how to migrate a typical 8903B program to the U8903A, using an FM receiver sensitivity test as an example. The test determines the lowest level of an RF signal that can be adequately received by a radio. For two-way radios, this is typically the lowest power that is able to achieve a SINAD value of 12 dB in the demodulated audio signal.

The receiver sensitivity test is usually performed by decreasing the RF input power to a radio and monitoring the SINAD of its audio output. The corresponding RF power level just before the SINAD falls below 12 dB gives the sensitivity of a two-way radio.

Table 7 contains 8903B and U8903A sample programs that perform the receiver sensitivity test. The programs control their respective audio analyzers to generate a sine wave stimulus into an E4438C RF signal generator, which will then modulate it to provide an RF input signal into the two-way radio. The audio output of the two-way radio is then fed back into the audio analyzer for SINAD measurements. The test setup is as shown in Figure 2.

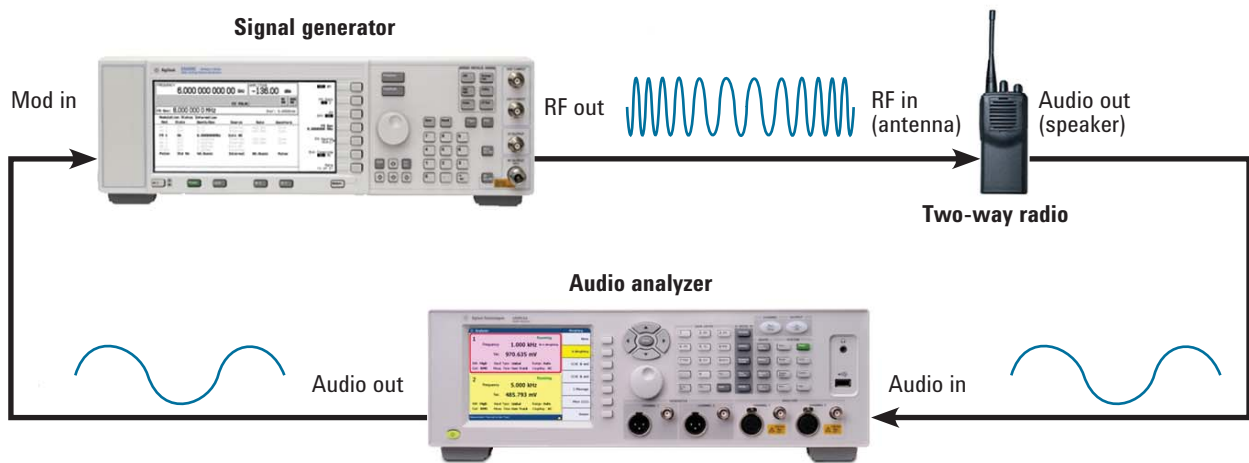


Figure 2. Typical receiver sensitivity test setup

Table 7. Receiver sensitivity test code migration

8903B program sequence	U8903A program sequence	Note
10 ASSIGN @Gpib TO 7	10 ASSIGN @Gpib TO 7	
20 ASSIGN @Aud TO 728	20 ASSIGN @Aud TO 728	
30 ASSIGN @Siggen TO 717	30 ASSIGN @Siggen TO 717	
40 CLEAR @Gpib	40 CLEAR @Gpib	
50 CLEAR SCREEN	50 CLEAR SCREEN	
60 !	60 !	
70 !	70 !	
80 ! **** Signal Generator Init ****	80 ! **** Signal Generator Init ****	
90 OUTPUT @Siggen;"FREQ 450 MHZ"	90 OUTPUT @Siggen;"FREQ 450 MHZ"	[1]
100 OUTPUT @Siggen;"POW -50 DBM"	100 OUTPUT @Siggen;"POW -50 DBM"	
110 OUTPUT @Siggen;"FM:DEV 20 KHZ"	110 OUTPUT @Siggen;"FM:DEV 20 KHZ"	
120 OUTPUT @Siggen;"FM:SOUR EXT1"	120 OUTPUT @Siggen;"FM:SOUR EXT1"	
130 OUTPUT @Siggen;"SOUR:FM:STAT ON"	130 OUTPUT @Siggen;"SOUR:FM:STAT ON"	
140 OUTPUT @Siggen;"OUTP:MOD ON"	140 OUTPUT @Siggen;"OUTP:MOD ON"	
150 OUTPUT @Siggen;"OUTP ON"	150 OUTPUT @Siggen;"OUTP ON"	
160 !	160 !	
170 !	170 !	
180 ! **** Audio Analyzer Init ****	180 ! **** Audio Analyzer Init ****	
190 OUTPUT @Aud;"AP6VL"	190 OUTPUT @Aud;"*CLS;*RST"	[2]
200 OUTPUT @Aud;"FR1KZ"	200 OUTPUT @Aud;"OUTP:STAT ON, (@1)"	
210 OUTPUT @Aud;"16.1SP"	210 OUTPUT @Aud;"SOUR:VOLT 6 VRMS, (@1)"	
220 OUTPUT @Aud;"M2"	220 OUTPUT @Aud;"SOUR:FREQ 1 KHZ, (@1)"	
230 OUTPUT @Aud;"T1"	230 OUTPUT @Aud;"SENS:FUNC2 SIN, (@1)"	
240 WAIT 1	240 WAIT 1	
250 !	250 !	
260 Rfpower=-75	260 Rfpower=-75	
270 Decrement=-.5	270 Decrement=-.5	
280 !	280 !	
290 !	290 !	
300 ! **** Loop ****	300 ! **** Loop ****	
310 REPEAT	310 REPEAT	
320 Rfpower=Rfpower+Decrement	320 Rfpower=Rfpower+Decrement	[3]
330 OUTPUT @Siggen;"POW ";Rfpower;" DBM"	330 OUTPUT @Siggen;"POW ";Rfpower;" DBM"	
340 WAIT .5	340 WAIT .5	
350 OUTPUT @Aud;"T2"	350 OUTPUT @Aud;"INIT:ANAL (@1);:FETC? FUNC2, (@1)"	
360 ENTER @Aud;Reading	360 ENTER @Aud;Reading	
370 PRINT "Input Power = ",Rfpower," dBm, SINAD = ",Reading," dB"	370 PRINT "Input Power = ",Rfpower," dBm, SINAD = ",Reading," dB"	
380 UNTIL Reading<12 OR Reading>9.E+9	380 UNTIL Reading<12 OR Reading>9.E+9	
390 !	390 !	
400 !	400 !	
410 PRINT "*****"	410 PRINT "*****"	
420 PRINT "The UUT Sensitivity is ";Rfpower- Decrement;" dBm"	420 PRINT "The UUT Sensitivity is ";Rfpower- Decrement;" dBm"	[4]
430 PRINT "*****"	430 PRINT "*****"	
440 END	440 END	

Notes

- The signal generator is set up to generate an FM modulated signal with a carrier frequency of 450 MHz. The modulating signal is from an external source.
- In the U8903A program, the \*CLS; \*RST commands perform the same function as the 8903B CLEAR @Gpib command. The OUTP:STAT ON, (@1) command needs to be always included. The audio analyzer is set up to generate a low distortion 6 V, 1 kHz sine wave to the modulating input of the signal generator. The audio analyzer is also configured to measure the SINAD at the radio audio output. Note that the 16.1SP command on the 8903B is used to set the SINAD resolution to 0.01 dB. This, and the T1 command to set the trigger hold mode, is not necessary for the U8903A.
- The signal generator RF power is decremented in steps of 0.5 dB, starting from -75 dBm. The audio analyzer SINAD reading is compared to 12 dB. If the reading is lower than 12 dB, the loop breaks. The loop will also break if the audio analyzer is unable to make a measurement.
- The sensitivity value is the RF power one step before the SINAD falls below 12 dB.



If you compare the execution times between the two programs, you will notice that the U8903A runs significantly faster. This is because of its faster command processing time. Actual test times will differ depending on the number of iterations.

When performing SINAD measurements with the U8903A, you will notice further enhancements. Each time the input frequency changes, the 8903B takes some time to tune to the new frequency. As the U8903A SINAD measurement is implemented through DSP, its speed is fixed no matter how quickly the fundamental frequency varies. Not only that, the U8903A is also able to make SINAD measurements at input levels a couple of orders of magnitude below the 50 mV threshold of the 8903B.

## Unsupported Commands

The 8903B commands that are fully supported by equivalent commands on the U8903A are listed in the Appendix. They cover all of the commands that are required to develop a typical test program. 8903B commands that are not supported on the U8903A are generally for features that are obsolete and therefore not supported on the U8903A. Examples include the X-Y plotter commands, redundant as the U8903A now has a graph plot for sweeps, and the commands to set post-notch filter gain — redundant as the U8903A notch filter is fully implemented by DSP. A peculiar 8903B command is `21.0SP`, which displays the GPIB address in binary format. This might be useful for instruments where the GPIB address is set with a DIP switch, but is otherwise redundant for modern instruments.

The command to set average detection type is also not supported by the U8903A, simply because this feature is not supported. Again, this feature might have been useful 20 years ago to correlate measurements with the average-responding voltmeters prevalent at the time, but it is not necessary nowadays as modern multimeters almost exclusively use true-rms detection.

The U8903A also does not have any equivalent for the 8903B `UP` and `DN` commands, which are used to increment and decrement the source frequency or amplitude. You can easily use iterative constructs such as the `FOR – NEXT` statements in HP BASIC to perform the same tasks. Similarly, the command to set ratio mode, `R1`, is not supported by the U8903A. However, calculating the ratio programmatically is not only simple in any

programming language, it is also faster compared to the ratio comparisons on the 8903B, which are performed electronically through hardware.

Finally, the `T1` command to set trigger hold in the 8903B is redundant for the U8903A. Once a remote command is sent to the U8903A, it automatically changes to trigger hold mode, as per the convention with modern SCPI-compliant instruments.

## Hints and Tips

When migrating a test program from one instrument to another, you need to consider their hardware differences as these might require workarounds. A good example is with the plug-in filters. As described previously, the U8903A allows up to three filters to be applied at a time, as opposed to two for the 8903B. This necessitates the insertion of code to turn off one filter on the U8903A just to ensure that not more than two filters are enabled at any one time.

Another case in point relates to AC/DC coupling. The 8903B input is fixed as DC coupled for the DC voltage measurement, while it is fixed as AC coupled for other measurements. The U8903A is more flexible — its inputs can be manually set to either AC or DC coupled.

The U8903A is AC coupled upon power-up, making it compatible with the 8903B. However, the U8903A does not allow you to change the measurement mode to DC voltage until you have set the coupling to DC. Therefore, if you are going to make a DC voltage measurement somewhere in your test program, you need to make sure the coupling is set to DC prior to making this measurement. After making the DC voltage measurement, you need to change the coupling back to AC for the other measurements, otherwise the readings may not correlate with the 8903B. This is because a signal with a significant DC offset will return different  $V_{rms}$  values depending on whether AC or DC coupling is set.

Due to this, a program that measures DC, then AC voltage in the 8903B would require a few more lines of code for the U8903A, as shown in Table 8.

Table 8. Changing coupling modes on the U8903A

8903B program sequence	U8903A program sequence	Note
ASSIGN @Aud TO 728	ASSIGN @Aud TO 728	
OUTPUT @Aud;"S1"	OUTPUT @Aud;"INP:COUP DC, (@1) "	[1]
OUTPUT @Aud;"T2"	OUTPUT @Aud;"SENS:FUNC2 VDC, (@1) "	[2]
ENTER @Aud;Dcv	ENTER @Aud;Dcv	
OUTPUT @Aud;"M1"	OUTPUT @Aud;"SENS:FUNC2 VAC, (@1) "	[3]
OUTPUT @Aud;"T2"	OUTPUT @Aud;"INP:COUP AC, (@1) "	[4]
ENTER @Aud;Acv	ENTER @Aud;Acv	

Notes

1. Sets DC voltage measurement mode. The U8903A program needs an additional command to set DC coupling mode before this.
2. Measures and reads DC voltage.
3. Sets AC voltage measurement mode. The U8903A program needs an additional command to revert to AC coupling mode after this.
4. Measures and reads AC voltage.

A major improvement in the U8903A programming is in results retrieval. The 8903B has a left and a right display. The left display shows frequency, while the right display shows any one of six measurements (Vdc, Vac, SINAD, Distortion, Distortion Level, or SNR). To retrieve the readings from both displays, you need to send an **RL** (read left) command, followed by an **RR** (read right) command. Note that once one of these commands is sent, it remains active for subsequent reads until the alternate command is sent.

A typical 8903B code sequence to read frequency and voltage would thus look like this:

```
OUTPUT @Aud;"RL"
OUTPUT @Aud;"T2"
ENTER @Aud;Freq
OUTPUT @Aud;"RR"
OUTPUT @Aud;"T2"
ENTER @Aud;Acv
```

The U8903A has the capability to fetch both results with a single query. Although you can still retrieve the readings separately, it makes more sense to write the code as shown below:

```
OUTPUT @Aud;"INIT:ANAL (@1);:FETC? ALL, (@1) "
ENTER @Aud;Freq,Acv
```

## Conclusion

Migrating your 8903B program to the U8903A should be relatively straightforward. You should be aware of the key hardware differences that necessitate minor changes, mostly by the insertion of additional lines of code. These have all been highlighted in this application note, and include the differences in maximum source frequency setting, input range, filter configuration, and AC/DC coupling.

Once you have successfully migrated your code to the U8903A, you will immediately appreciate its benefits — above all for its faster measurement times and friendlier user interface. The fact that the U8903A can do much more, while protecting your investment in your legacy test system, makes it an ideal replacement for the 8903B.

# Appendix

Table A-1

Generator commands	
8903B command	Equivalent U8903A SCPI command
AP<voltage>{VL MV DV}	SOURce:VOLTage <voltage>[VRMS MVRMS DBV], (@1)
FR<freq>{HZ KZ}	SOURce:FREQuency <freq>[HZ KHZ], (@1)
47.0SP	OUTPut:IMPedance IMP50, (@1)
47.1SP	OUTPut:IMPedance IMP600, (@1)
10.0SP	DISPlay:VIEW GEN2,CHAN1
Measurement results retrieval commands	
8903B command	Equivalent U8903A SCPI command
T2	INITiate:ANALyzer (@1)::FETCh? FUNC1, (@1) or INITiate:ANALyzer (@1)::FETCh? FUNC2, (@1)
T3	INITiate:ANALyzer (@1)::FETCh? FUNC1, (@1) or INITiate:ANALyzer (@1)::FETCh? FUNC2, (@1)
Analyzer commands	
8903B command	Equivalent U8903A SCPI command
M1	SENSe:FUNCTion2 VAC, (@1)
M2	SENSe:FUNCTion2 SINad, (@1)
M3	SENSe:FUNCTion2 THDRatio, (@1)
S1	INPut:COUpling DC, (@1) and SENSe:FUNCTion2 VDC, (@1)
S2	SENSe:FUNCTion2 SNRatio, (@1)
S3	SENSe:FUNCTion2 THDLevel, (@1)
AU	SENSe:VOLTage:RANGE:AUTO ON, (@1)
1.0SP (Auto)	SENSe:VOLTage:RANGE:AUTO ON, (@1)
1.1SP (300 V)	Not Available
1.2SP (189 V)	Not Available
1.3SP (119 V)	SENSe:VOLTage:RANGE 140V, (@1)
1.4SP (75.4 V)	SENSe:VOLTage:RANGE 100V, (@1)
1.5SP (47.6 V)	SENSe:VOLTage:RANGE 50V, (@1)
1.6SP (30.0 V)	SENSe:VOLTage:RANGE 50V, (@1)
1.7SP (18.9 V)	SENSe:VOLTage:RANGE 25V, (@1)
1.8SP (11.9 V)	SENSe:VOLTage:RANGE 12.8V, (@1)
1.9SP (7.54 V)	SENSe:VOLTage:RANGE 12.8V, (@1)
1.10SP (4.76 V)	SENSe:VOLTage:RANGE 6.4V, (@1)
1.11SP (3.00 V)	SENSe:VOLTage:RANGE 3.2V, (@1)
1.12SP (1.89 V)	SENSe:VOLTage:RANGE 3.2V, (@1)
1.13SP (1.19 V)	SENSe:VOLTage:RANGE 1.6V, (@1)
1.14SP (0.754 V)	SENSe:VOLTage:RANGE 0.8V, (@1)
1.15SP (0.476 V)	SENSe:VOLTage:RANGE 0.8V, (@1)
1.16SP (0.300 V)	SENSe:VOLTage:RANGE 0.4V, (@1)
1.17SP (0.189 V)	SENSe:VOLTage:RANGE 0.4V, (@1)
1.18SP (0.119 V)	SENSe:VOLTage:RANGE 0.4V, (@1)
1.19SP (0.0754 V)	SENSe:VOLTage:RANGE 0.4V, (@1)
2.0SP (Auto)	SENSe:VOLTage:RANGE:AUTO ON, (@1)
2.1SP (300 V)	Not Available
2.2SP (64 V)	SENSe:VOLTage:RANGE 100V, (@1)
2.3SP (16 V)	SENSe:VOLTage:RANGE 25V, (@1)
2.4SP (4 V)	SENSe:VOLTage:RANGE 6.4V, (@1)

A0	(RMS)	SENSe:VOLTage:DETEctor RMS, (@1)
5.0SP	(Fast RMS)	SENSe:VOLTage:DETEctor RMS, (@1)
5.1SP	(Slow RMS)	SENSe:VOLTage:DETEctor RMS, (@1)
5.7SP	(Quasi-peak)	SENSe:VOLTage:DETEctor QPK, (@1)
H0		SENSe:FILTer:HPASs NONE, (@1);WEIGHting NONE, (@1)
H1		SENSe:FILTer:HPASs HP400, (@1) or SENSe:FILTer:WEIGHting {AWEIGHTing CCIR1k CCIR2k CMESsage CCITt}, (@1)
H2		SENSe:FILTer:HPASs HP400, (@1) or SENSe:FILTer:WEIGHting {AWEIGHTing CCIR1k CCIR2k CMESsage CCITt}, (@1)
L0		SENSe:FILTer:LPASs NONE, (@1)
L1		SENSe:FILTer:LPASs LP30, (@1)
L2		SENSe:FILTer:LPASs CUSTom, (@1)
9.0SP		SENSe:VOLTage:RANGE:AUTO OFF, (@1)
19.0SP		SENSe:REFerence:IMPedance 8, (@1);FUNCTion2:UNIT W, (@1)
19.NNNSP		SENSe:REFerence:IMPedance <impedance>, (@1);FUNCTion2:UNIT W, (@1)

### Sweep commands

8903B command	Equivalent U8903A SCPI command
FA<start>	SOURce:SWEep:STArT <start>, (@1)
FB<stop>	SOURce:SWEep:STOP <stop>, (@1)
17.0SP (0/dec)	SOURce:SWEep:POINts <points>, (@1)
17.1SP (1/dec)	
17.2SP (2/dec)	
17.3SP (5/dec)	
17.4SP (10/dec)	
17.5SP (20/dec)	
17.6SP (50/dec)	
17.7SP (100/dec)	
17.8SP (200/dec)	
17.9SP (500/dec)	
W1	INITiate:SWEep

### System commands

8903B command	Equivalent U8903A SCPI command
21.1SP	SYSTem:COMMunicate:GPIB:ADDRess?
22.1SP	*SRE 16
22.2SP	*ESE 32; *SRE 32
22.4SP	*SRE 4



### Agilent Email Updates

[www.agilent.com/find/emailupdates](http://www.agilent.com/find/emailupdates)  
Get the latest information on the products and applications you select.



### Agilent Direct

[www.agilent.com/find/agilentdirect](http://www.agilent.com/find/agilentdirect)  
Quickly choose and use your test equipment solutions with confidence.



[www.agilent.com/find/open](http://www.agilent.com/find/open)  
Agilent Open simplifies the process of connecting and programming test systems to help engineers design, validate and manufacture electronic products. Agilent offers open connectivity for a broad range of system-ready instruments, open industry software, PC-standard I/O and global support, which are combined to more easily integrate test system development.

## Remove all doubt

Our repair and calibration services will get your equipment back to you, performing like new, when promised. You will get full value out of your Agilent equipment throughout its lifetime. Your equipment will be serviced by Agilent-trained technicians using the latest factory calibration procedures, automated repair diagnostics and genuine parts. You will always have the utmost confidence in your measurements. For information regarding self maintenance of this product, please contact your Agilent office.

Agilent offers a wide range of additional expert test and measurement services for your equipment, including initial start-up assistance, on-site education and training, as well as design, system integration, and project management.

For more information on repair and calibration services, go to:

[www.agilent.com/find/removealldoubt](http://www.agilent.com/find/removealldoubt)

Product specifications and descriptions in this document subject to change without notice.

For more information on Agilent Technologies' products, applications or services, please contact your local Agilent office. The complete list is available at:

[www.agilent.com/find/contactus](http://www.agilent.com/find/contactus)

### Americas

Canada	(877) 894-4414
Latin America	305 269 7500
United States	(800) 829-4444

### Asia Pacific

Australia	1 800 629 485
China	800 810 0189
Hong Kong	800 938 693
India	1 800 112 929
Japan	0120 (421) 345
Korea	080 769 0800
Malaysia	1 800 888 848
Singapore	1 800 375 8100
Taiwan	0800 047 866
Thailand	1 800 226 008

### Europe & Middle East

Austria	01 36027 71571
Belgium	32 (0) 2 404 93 40
Denmark	45 70 13 15 15
Finland	358 (0) 10 855 2100
France	0825 010 700*
	*0.125 €/minute
Germany	07031 464 6333**
Ireland	1890 924 204
Israel	972-3-9288-504/544
Italy	39 02 92 60 8484
Netherlands	31 (0) 20 547 2111
Spain	34 (91) 631 3300
Sweden	0200-88 22 55
Switzerland	0800 80 53 53
United Kingdom	44 (0) 118 9276201

Other European Countries:

[www.agilent.com/find/contactus](http://www.agilent.com/find/contactus)

Revised: October 1, 2008

© Agilent Technologies, Inc. 2009  
Printed in USA, June 29, 2009  
5990-4135EN



Agilent Technologies