Increase Power Amplifier Test Throughput with the Agilent M9381A PXIe Vector Signal Generator
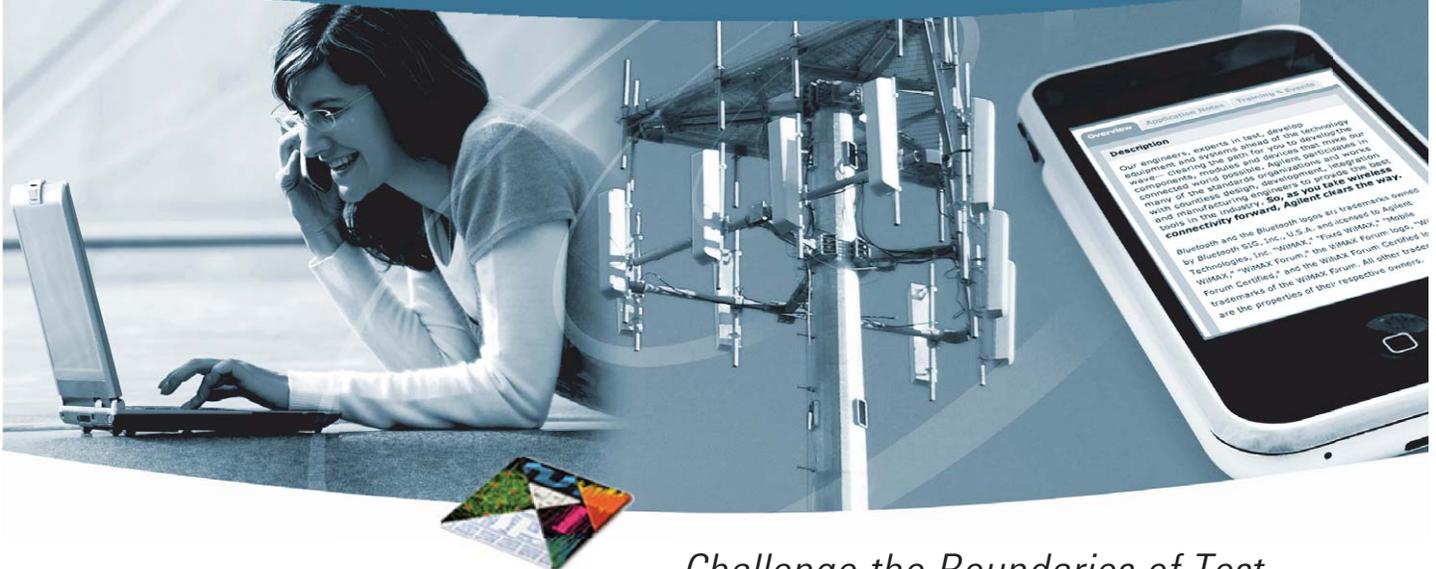
Challenge the Boundaries of Test
Agilent Modular Products

Accelerate power amplifier test throughput with the world's fastest PXIe Vector Signal Generator. Achieve cost reductions in test while maintaining high test quality.

Power amplifier designs for mobile handsets are becoming more complex, which directly impacts test demands and the cost of test. Complexity increases with the introduction of new, wider bandwidth standards and the addition of the number of radios to each device. Business issues, such as pressure on prices of these devices places greater demands on engineering teams producing power amplifiers.

Engineers who test mobile power amplifiers and front end modules are looking for ways to reduce test cost through maximizing throughput while ensuring that the devices meet required performance levels.

This application note will discuss these complex issues and recommended solutions.

Anticipate ___Accelerate ___Achieve

Agilent Technologies

OVERVIEW

Introduction

This application note provides an overview of the key issues in a power amplifier and front end module test system related to the RF signal generator. Further, the note offers a solution to achieve fast test throughput using the modular Agilent M9381A PXIe Vector Signal Generator (VSG).

Contents

Your Benefits

Accelerate test throughput. Agilent's exclusive baseband tuning technology innovation enables frequency and amplitude switching speeds as fast as 10 µs in list mode and 250 µs from its programming interface.

Meet current and anticipate future test needs. High output power up to +18 dBm across the 1 MHz to 6 GHz frequency range, excellent linearity and level accuracy (± 0.4 dB), and outstanding adjacent power ratio performance for output levels up to +10 dBm with little degradation.

Achieve lower cost of ownership. In additional to Agilent's well known worldwide support, factory warranty and calibration, Agilent also provides easy product integration into your test system, interoperability using Agilent Connection Expert and license key upgradable options.

Figure 1. Agilent M9381A PXIe VSG (includes M9311A, M9310A and M9301A) pictured with M9300A frequency reference, M9036A embedded controller in a M9018A 18-slot chassis.

Key Issues Faced by Power Amplifier Test Engineers

This application note addresses test challenges faced by power amplifier engineers including:

- The need to quickly adjust power level inputs to the device under test (DUT).

- The need to assess modulation performance (i.e., ACPR and EVM) at high VSG output levels.

The figure below shows a simplified block diagram for the RF vector signal generator and analyzer in a typical power amplifier/front end module test system.

Typical power amplifier modules require an input power level of 0 to + 5 dBm, digitally modulated according to communication standards such as WCDMA or LTE. The specified performance of the power amplifier or front end module is normally set at a specific output level of the DUT.   If the devices have small variations in gain, it may be necessary to adjust the power level from the VSG to get the correct output level of the DUT. Only after the DUT output level is set at the correct value can the specified parameters be tested. The time spent adjusting the VSG to get the correct DUT output power can be a major contributor to the test time and the overall cost of test.

The VSG is connected to the DUT using a cable and switches. The switching may be used to support testing of multi-band modules or multi-site testing. The complexity of the switching depends on the number of bands in the devices and the number of test sites supported by the system. The DUTs are typically inserted into the test fixture using an automated part hander. In some cases, several feet of cable is required between the VSG and the input of the DUT.

The combination of the RF cables and the switching network can add several dB of loss between the output of the VSG and the input of the DUT, which requires higher output levels from the VSG. Since the tests are performed with a modulated signal, the VSG must also have adequate modulation performance at the higher power levels.
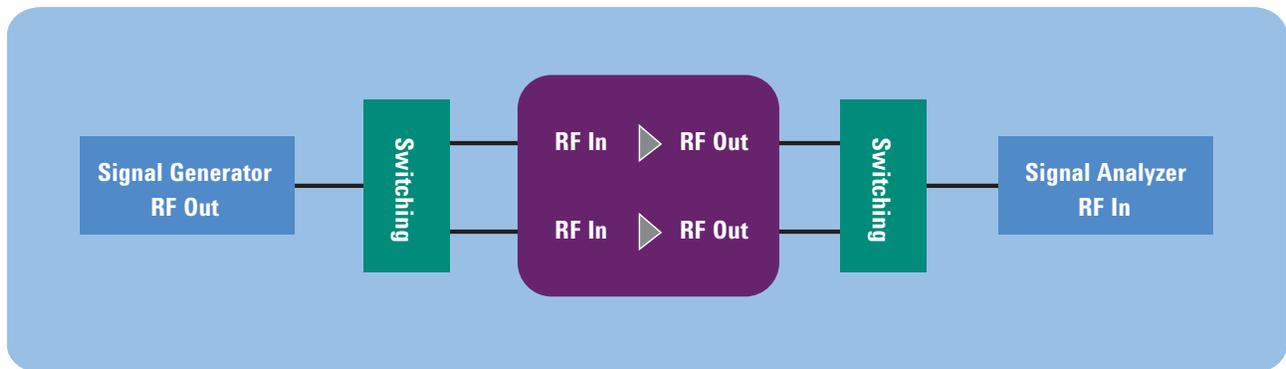

Figure 2. Typical Power Amplifier / Front End Module Test Setup.

Using the M9381A to Increase Test Throughput and Quality

The Agilent M9381A PXIe VSG offers unique features that:

- Increase test throughput with fast amplitude and frequency switching.
- Provide good modulation performance, particularly at high power levels.
- Provide very linear power level changes.

The M9381A reduces the overall switching time through a powerful, innovative tuning methodology. It further increases throughput by providing good linearity and repeatability which reduces the number of iterations required to get the DUT to the correct power output level. The M9381A also offers high modulation quality to drive amplifiers directly without adding external amplification.

Several other VSGs offer amplitude and frequency switching times of less than 1 ms when used in list modes. However, since the output level of the VSG cannot be predetermined for each test, list modes cannot be used for power amplifier testing. Other VSGs require significantly longer switching time when controlled through a normal programming interface. The Agilent M9381A offers the fastest switching time on the market of 250 µs from its programming interface and 10 µs in list mode, using a baseband tuning innovation.

The M9381A comes with 40 MHz bandwidth, but can be upgraded to 100 MHz or 160 MHz. The baseband frequency offset can be programmed to any offset within the purchased modulation bandwidth. For example, the 160 MHz bandwidth option allows the baseband frequency offset to be set to ±80 MHz. The baseband power offset can be set to 20 dB below the programmed RF power level and still achieves high quality modulation performance. To take advantage of this feature for power amplifier testing, engineers can set the RF frequency to the center of the band being tested and the RF power level to the maximum required for all tests. From there, baseband frequency adjustments are made to test at multiple frequencies across the band and the baseband power level is adjusted to servo the DUT output level to the correct value.

Better linearity, repeatability and resolution offered by the Agilent M9381A further reduces the test time by enabling the servo loop to converge in fewer steps. After the DUT output level is measured by the signal analyzer, the new value of the VSG output power is calculated based on the difference between the measured power and the desired power. Then, the VSG is adjusted by the amount necessary to achieve the correct DUT output power.

Figure 3, below, shows the measured linearity of the M9381A, over a 20 dB range from + 10 dBm to -10 dBm, using both the baseband and RF level adjustments, compared to a different PXI VSG.

Additionally, the M9381A provides high modulation quality, particularly at high output power levels. In many cases, signal levels as high as + 15 dBm may need overcome the loss between the VSG and the DUT input. As shown in Figure 4 below, the M9381A has excellent adjacent channel power ratio (ACPR) at high output power levels. At +10 dBm, there is little or no degrading of the ACPR and at +15 dBm, the ACPR level is still near 60 dBc.

The M9381A can be used with Agilent's Signal Studio signal creation software. Its performance optimized reference signals enhance the characterization and verification of devices. Signals include wireless standards such as LTE, WCDMA and 802.11ac.
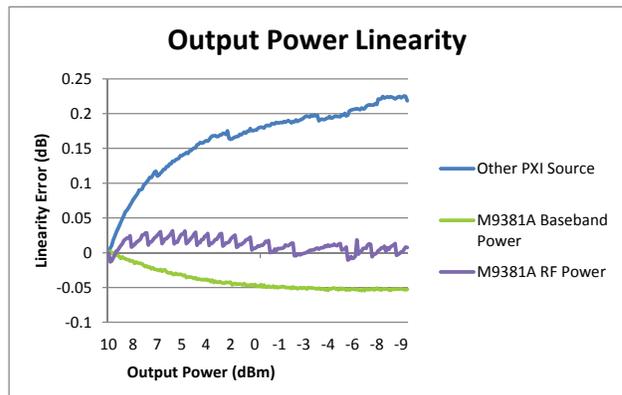

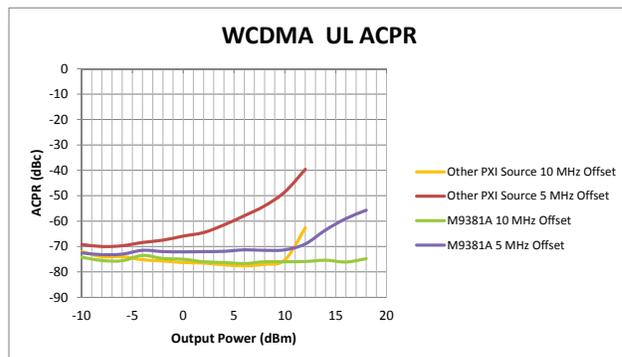Figure 3. Note M9381A linearity error within ±0.05 dBm range.

Figure 4. Note the M9381A shows little ACPR degradation below 10 dBm and is better than 60 dBc at 15 dBm.

Solution Details

This section will provide an overview of programming the M9381A using Agilent supplied drivers. It also provides detailed information for programming the baseband tuning method described above.

Overview of M9381A Drivers

The M9381A driver installation package contains software required to program the M9381A in a variety of Application Development Environments (ADE). The driver installation package can be downloaded from the Agilent website at www.agilent.com/find/m9381a. Agilent I/O Libraries suite (version 16.2) is required prior to installing the M9381A driver. Please view the webpage and download instructions at www.agilent.com/find/iosuite.

Agilent PXIe modules use two primary drivers: an IVI-COM driver and an IVI-C driver, allowing programming across a wide range of Microsoft Windows® ADEs. In addition to the C and COM drivers, LabVIEW and MATLAB drivers are also provided. The driver installation program installs the LabVIEW driver in the appropriate instr.lib directory for all versions of LabVIEW greater than 8.5 installed on the PC. The installation procedure also integrates the LabVIEW driver help into the LabVIEW help system and the IVI driver help into the Microsoft Visual Studio® help system, providing programming assistance in those ADEs. A Windows help file is also installed and can be accessed from the Windows start menu at:

All Programs → Agilent IVI Drivers →M938x Source →Documentation

The help file includes a description of ways to include, initialize, and use the C and COM drivers in C, C++, Visual Basic and C#. The driver installation program also installs example programs that can be found in the IVI Foundation folder at Drivers/M938x/Examples. The location of the IVI Foundation folder will vary between 32 and 64 bit operating systems. For 64 bit systems, both 32 and 64 bit drivers will be installed.

Programming the M9381A for Power Amplifier and Front End Module Test Systems

To maximize throughput, the programming of the M9381A should be split into three general categories:

Initialization. The following should be executed one time when the test system initializes:

- Initialization of the M9381A driver
- Loading of any arbitrary waveform data and files
- Creation of any lists or ARB sequences

Execution. This code programs the VSG to the correct state for each test and is repeated for each device tested. There are a number of commands that will be used in this section. The details are provided below.

Finalization. This code will be executed one time before the test system software exits. In this code the IVI driver should be closed. When the driver is closed, the output of the source will be disabled. The process of opening the IVI driver takes a few seconds, so the driver should be left open until exiting the test program to achieve maximum test throughput.

M9381A Soft Front Panel	MATLAB	Visual Studio (VB.Net, C#)		Visual Studio (C/C++)	LabVIEW	LabWindows/CVI
	MATLAB Driver	Agilent VEE			LabVIEW Driver	
		NI TestStand				
IVI-COM Drivers				IVI-C Drivers		

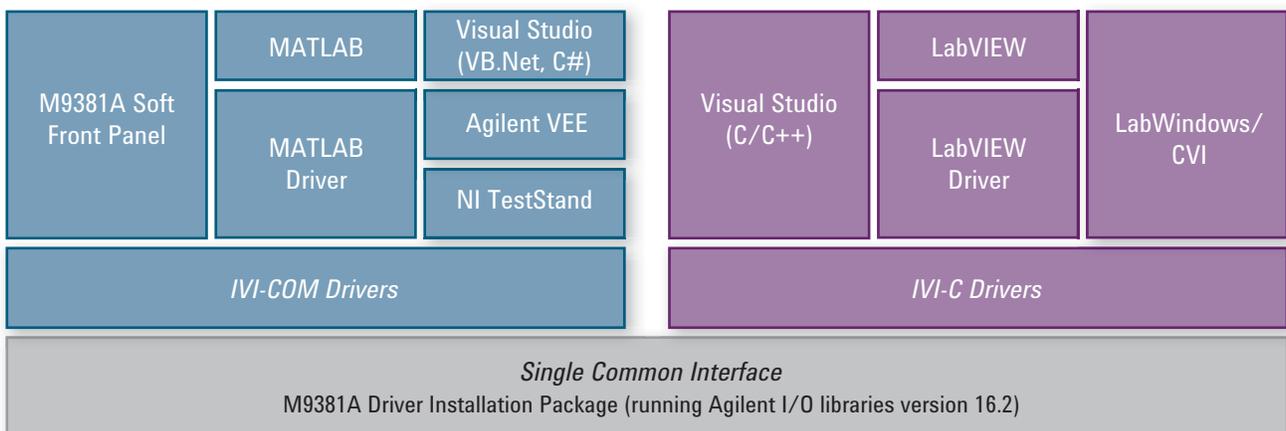| Single Common Interface |
|---|
| M9381A Driver Installation Package (running Agilent I/O libraries version 16.2) |

Figure 5. Driver software installed for the M9381A VSG in addition to the instrument library and Agilent I/O Libraries.

Initializing the IVI Driver

The initialization method for the IVI driver requires a resource string parameter to identify the addresses of the M9381A hardware. The parameter can be of two formats. The first format is a string of the module VISA addresses separated by a ";" in the Selected Instrument window as shown in Figure 6, below.
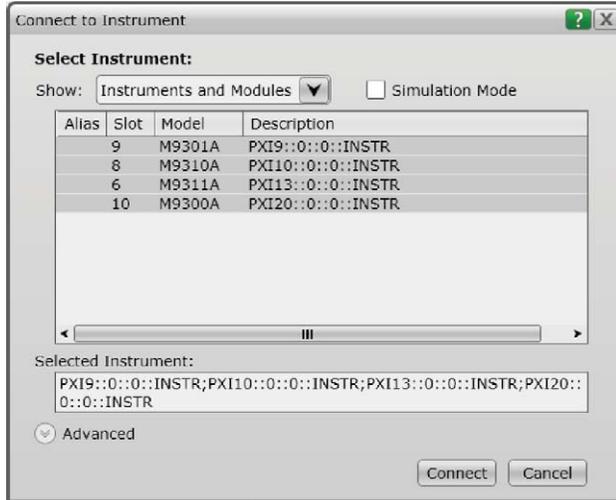

Figure 6. M9381A soft front panel connecting to hardware.

The second format is a string that matches an IVI configuration store instrument connection that was saved using the M9381A Soft Front Panel (SFP) as shown in Figure 7 using the File → Save Connection menu selection.
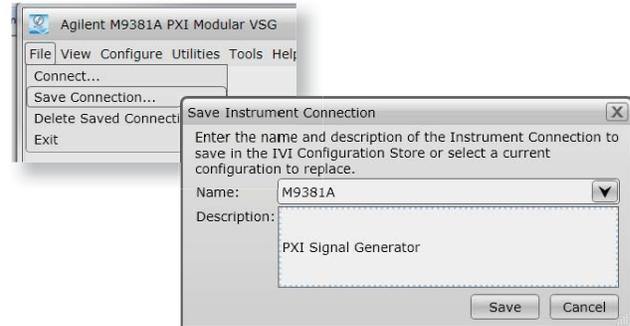

Figure 7. M9381A soft front panel saving connection name.

The IVI driver can also be initialized in simulation mode, allowing development and testing of the software without a connection to the M9381A hardware. Figure 8, below, shows a Visual Basic code example for initializing the driver, in hardware and simulation modes.

```
5        ' Create an unitialized instance of the driver.
6        Dim driver As AgM938x = Nothing
7
8        ' Initialize driver in simulation or hardware mode
9        Dim simulate As Boolean = False
10       Try
11           If simulate = True Then
12               ' Show the IVI Configuration Store resource string format here
13               Dim resource As String = "M9381A"
14               Dim options As String = "QueryInstrStatus=true, Simulate=true, DriverSetup= Model=, Trace=false"
15               driver.Initialize(resource, True, True, options)
16           Else
17               ' Show the VISA address resource string format here
18               Dim resource As String = "PXI12::0::0::INSTR;PXI11::0::0::INSTR;PXI8::0::0::INSTR;PXI13::0::0::INSTR"
19               Dim options As String = "QueryInstrStatus=true, Simulate=false, DriverSetup= Model=, Trace=false"
20               driver.Initialize(resource, True, True, options)
21           End If
22
```

Figure 8. Visual Basic code example for initialization in hardware and simulation modes.

Loading Waveform Data

The M9381A has options for up to 1 giga sample of wave-form playback memory, allowing multiple waveform files to be loaded into memory at the same time. The M9381A driver provides several interfaces for loading waveforms into the VSG. Waveforms that are created with the Agilent Signal Studio applications can be directly loaded into the M9381A, allowing users to use the same modulation data files with the M9381A and other Agilent signal generators, such as the N5182B MXG signal generator. For Signal Studio files, the API simply needs a reference name for the waveform and full path to the file.

Signal Studio waveform files can be copied from the internal disk of other Agilent Signal generators or created directly in Signal Studio. To create the waveforms in Signal Studio, launch the Signal Studio Application, config-ure the waveform parameters using the UI of by loading a settings file and then select File | Export Waveform Data... from the menu, as shown in Figure 9.

Binary data files with interleaved I and Q data in single or double precision floating point and 16 bit integer formats can also be loaded directly into the M9381A. There are also methods in the IVI driver to load the same data formats from arrays of data, allowing the user to import data from other file formats or to generate the waveform data directly in the application program. In addition to the waveform data, the M9381A needs to be provided with the sample rate, RMS or peak to average value and the DAC factor. These values are included in the header of Signal Studio files. When loading binary data, these values need to be provided to the API.
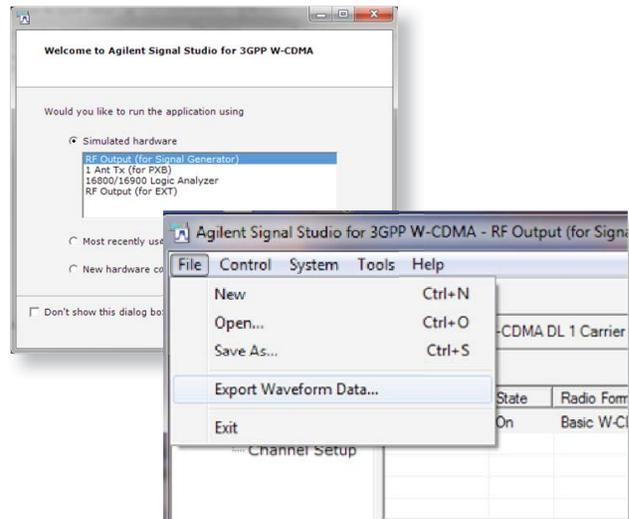

Figure 9. Saving a waveform file from Signal Studio.

Waveforms can also contain marker data to blank the RF output control when the ALC tracks the signal level. The marker data also provides trigger signals on the M9381A front panel trigger connections. For Signal Studio files, the marker data and assignment of markers to pulse and ALC blanking is included in the file and is automatically imported. Binary data in double precision floating point format can be loaded with markers, both from a binary data file and from arrays of data.

Figure 10, below shows a Visual Basic code example for loading Signal Studio and Binary waveform data files; and for creating waveform and marker data in the application program:

- Line 24 of the code in Figure 10 loads a Signal Studio waveform file into the VSG.

- Lines 26-33 setup the information to load a binary file into the VSG. Note with the Signal Studio file, the additional parameters, including sample rate, peak to average ratio and percent of DAC scale do not need to be specified as they are included in the Signal Studio file header. For other waveform data sources, these parameters do need to be specified.

- Lines 35-63 generate and load waveform and marker data into the VSG. In this example the marker data is set high during the pulse and low at other times. This will cause the pulse modulator to turn off, increasing the on/off ratio of the pulsed signal and cause the ALC to hold when the pulse is not present.

```
22
23          ' Load a Signal Studio Waveform file
24          driver.Modulation.IQ.UploadArbAgilentFile("WCDMA_UL", "C:\\Waveforms\\WCDMA_UL.wfm")
25
26          ' Load a binary waveform file
27          ' Need to specify Sample Rate, RMS Power Level, DAC Scale Value, RF Blanking Marker and ALC Blanking Marker
28          Dim MySampleRate As Double = 20000000.0
29          Dim MyRmsPower As Double = 0              ' The RMS value (Peak to Average Ratio) can be calculated if not known
30          Dim MyScaleFactor As Double = 0.9        ' Scale factor is 0 to 1 choos value to avoid waveform clipping
31          driver.Modulation.IQ.UploadArbBinaryFile("BinFile", "C:\\Waveforms\\Myfile.bin", AgM938xBinaryArbEnum.AgM938xBinaryArbDP, _
32                                         MySampleRate, MyRmsPower, MyScaleFactor, _
33                                         AgM938xMarkerEnum.AgM938xMarkerNone, AgM938xMarkerEnum.AgM938xMarkerNone)
34
35          ' Create a waveform of Doubles and then load data
36          MySampleRate = 20000000.0
37          MyRmsPower = 0                  ' The RMS value (Peak to Average Ratio) can be calculated if not known
38          MyScaleFactor = 0.9            ' Scale factor is 0 to 1 choos value to avoid waveform clipping
39          Dim pulsePoints As Integer = 500
40          Dim points As Integer = 1000
41          Dim ArbData(points * 2 - 1) As Double
42          Dim MarkerData((points - 1)) As Byte
43          Dim PulseThreshold As Double = 0
44          ' Marker data will be LSB (marker 1) + next bit (Marker 2) = 1 + 2 = 3
45          Dim MarkerValue As Byte = 3
46
47          ' Use 1 for the pulse
48          For i = 0 To pulsePoints * 2 - 1 Step 2
49              ' Put pulse on I channel
50              ArbData(i) = 1
51              ArbData(i + 1) = 0
52              MarkerData(i / 2) = MarkerValue
53          Next
54          ' Use 0 for off time
55          For i = pulsePoints * 2 To points - 1 Step 2
56              ' I and Q both zero when pulse is off
57              ArbData(i) = 0
58              ArbData(i + 1) = 0
59              MarkerData(i / 2) = 0
60          Next
61
62          driver.Modulation.IQ.UploadArbDoublesWithMarkers("MyPulseData", ArbData, MarkerData, MySampleRate, MyRmsPower, MyScaleFactor, _
63                                         AgM938xMarkerEnum.AgM938xMarker1, AgM938xMarkerEnum.AgM938xMarker1)
64
```

Figure 10. Visual Basic code example for creating and loading waveform data.

Creating Lists and ARB Sequences

As previously discussed, testing of power amplifiers and front end modules typically prevent the use of list modes in signal generators since the output level of the signal generator varies for each test condition. The primary motivation for using list and sequence modes is to reduce the overall test time by eliminating software overhead from the control of the DUT and the test equipment. The M9381A supports robust and flexible list and sequence modes. However, this topic will not be covered in this application note as it does not typically apply to power amplifier testing. The M9381A baseband tuning mode can be used to provide fast tuning and is very well suited to power amplifier testing. The use of the baseband tuning mode will be discussed in detail in the following section.

Adjusting Frequency and Output Power

While testing a power amplifier or front end module, up to 40% of VSG test time is spent changing the VSG frequency and output levels. The tuning time for most VSGs is substantially longer when it is not used in list mode. The

M9381A offers baseband tuning methodology that delivers non-list mode switching times as fast as the list mode tuning times in many VSGs over the 1 MHz to 6 GHz frequency range.

The M9381A offers two modes for adjusting frequency and amplitude. RF tuning allows the VSG to be set across the complete operating frequency and amplitude range. Baseband tuning allows the frequency to be adjusted within the modulation bandwidth and up to a 20 dB reduction of the output level. These modes can be used together to provide the fastest throughput for power amp tests.

Figure 11, below shows a simplified Visual Basic example for a power amplifier input power adjustment loop for three frequencies. The example code in Figure 11 is simplified to highlight the programming of the M9381A VSG. It does not include programming for a signal analyzer or error checking to ensure that parameters do not go outside of allowable ranges or to force an exit from the while loop due to an unexpected condition.

```
65          ' Setup the VSG to output the WCDMA UL ARB file at the center of the frequency band
66          ' Set the power level to the maximum expected power and set the ALC off to use the
67          ' baseband power feature
68
69          Dim RfFrequency() As Double = {824500000.0, 836500000.0, 869500000.0}
70          Dim CenterFrequency As Double = RfFrequency(1)
71
72          Dim MaxPowerLevel As Double = 10.0
73          Dim DutOutputLevel As Double = 22.5 ' Power out of DUT for tests
74          Dim DutOutputTolerance As Double = 0.1 ' allowable error in DUT output level
75
76          driver.ALC.Enabled = False
77          driver.Modulation.Enabled = True
78          driver.RF.Configure(CenterFrequency, MaxPowerLevel)
79          ' Set starting power level to middle of range
80          driver.Modulation.BasebandPower = -10
81          driver.Modulation.PlayArb("WCDMA_UL", AgM938xStartEventEnum.AgM938xStartEventImmediate)
82
83          For Each CurrentFrequency As Double In RfFrequency
84              ' Set Frequency offset use previous power level as starting point
85              driver.Modulation.BasebandFrequency = CurrentFrequency - CenterFrequency
86              driver.Apply()
87              Dim MeasuredPower As Double
88              ' MeasuredPower = value measured from VSA
89              While Math.Abs(DutOutputLevel - MeasuredPower) > DutOutputTolerance
90                  ' Adjust Source
91                  driver.Modulation.BasebandPower = driver.Modulation.BasebandPower + _
92                      (DutOutputLevel - MeasuredPower)
93                  driver.Apply()
94                  'Measure power level again
95                  ' MeasuredPower = value measured from VSA
96              End While
97              ' At Correct output power now
98              ' Measure EVM, ACPR,etc.
99          Next
```

Figure 11. Visual Basic code example for power adjustment loop.

9

The M9381A drivers do perform range checking on parameters and will throw an exception if the values are outside of the allowable range:

- Lines 69-74 of the code in Figure 4 initialize the values for the three frequencies to be tested, the maximum output level of the source, the power level required at the DUT output and the allowable error in that power level.

- Lines 76-80 set the initial configuration of the VSG. Note that the ALC is disabled to allow use of the baseband power command, the RF frequency is set to the center of the band and the RF power level is set to the maximum expected power. The baseband offset is also set to an initial value. This value should be as close as possible to the expected level that will have the DUT at the correct output level. This value could be based on previously characterized data on the DUT or the value of the last DUT tested.

- Lines 76-80 send changes to the driver which do not take effect immediately. The changes are sent to the hardware by sending an "Apply" command, as in line 86 or by sending a "PlayArb" command, as in line 81. This allows for maximum throughput by combining all of the hardware changes together and only having to wait for the source hardware one time. In the above example, after line 81 is executed, the WCDMA waveform will be playing and the source will be at the commanded frequency and amplitude.

- Lines 83-99 demonstrate an adjustment loop for setting the DUT output level, then making the required parametric measurements and repeating this for each frequency in the band.

- Lines 85 and 86 set the baseband frequency offset and sends that setting to the VSG with the apply command. Notice that the value of the baseband frequency offset is relative to the RF frequency. Thus, for values less than the RF frequency, the value will be negative.

- Line 88 reads the DUT output power level and is commented out in this example.

- Lines 89-96 make up the DUT output level adjustment loop. This while loop will exit when the measured DUT output level is within the specified tolerance of the required level. Inside the loop the VSG baseband power level is set to the previous level plus the previous error and then the output level is measured again. If the DUT is in compression, a different algorithm may be needed to calculate the new VSG output level.

Once the program reaches line 98, the DUT output power is at the correct level and the parametric measurements begin.

For multi-band devices this section of code would be repeated for each band. The maximum modulation bandwidth option of the M9381A is 160 MHz. With this option it may be possible to test multiple bands of the DUT using the baseband frequency offset. If the bands of the DUT are further apart that the modulation bandwidth of the M9381A, the RF frequency can be changed for each band, as in line 78 of Figure 11 on the previous page.

Finishing the Program

Figure 12, below shows the remainder of the Visual Basic example program.

Lines 101-105 contain the exception handler for the "Try Catch" block started in line 10 of Figure 8 on page 6. Lines 108-115 include code to close the driver before the program exits. Notice that the close command is called outside of the "Try" block to insure that it is always called. The driver should always be closed before exiting the test program. For maximum throughput the driver should be opened at the beginning of the program and not closed until the end of the program.

```
100
101        Catch ex As Exception
102            ' Handle any exception
103            Console.WriteLine("Exception from M938x Driver:")
104            Console.WriteLine(ex.Message)
105            Console.ReadLine()
106        End Try
107
108        If Not driver Is Nothing Then
109            If driver.Initialized Then
110                ' Close the driver
111                driver.Close()
112                Console.WriteLine()
113                Console.WriteLine("Driver Closed")
114            End If
115        End If
116    End Sub
```

Figure 12. Visual Basic code example for error handling and closing the driver.

Summary

In summary, the M9381A combines the fastest signal switching speed on the market with excellent parametric performance and programming capability to accelerate power amplifier test throughput. It provides this in a PXI format allowing for a small test station footprint backed by Agilent support, calibration and warranties.

Ordering information

Model	Description
M9381A	PXIe Vector Signal Generator: 1 MHz to 3 or 6 GHz Includes: M9301A PXIe Synthesizer M9310A PXIe Source Output M9311A PXIe Digital Vector Modulator One day start up assistance Module interconnect cables Software, example programs and product information on CD Return to Agilent Warranty—3 Years
M9381A-300 Required for warranted specifications	PXIe Frequency Reference: 10 MHz and 100 MHz Adds M9300A PXIe Frequency Reference: 10 MHz and 100 MHz (M9300A module has five 100 MHz outputs and can support up to four M9381A VSGs in a single chassis.

Configurable Options	
✓ *Recommended configuration for power amplifier test application*	
Frequency	
M9381A-F03	Frequency range: 1 MHz to 3 GHz
✓ M9381A-F06	Frequency range: 1 MHz to 6 GHz
Power	
✓ M9381A-1EA	High Output Power
Switching Speed	
✓ M9381A-UNZ	Fast Switching
Modulation Bandwidth	
M9381A-B04	RF modulation bandwidth: 40 MHz
✓ M9381A-B10	RF modulation bandwidth: 100 MHz
M9381A-B16	RF modulation bandwidth: 160 MHz
Memory	
M9381A-M01	Memory, 32 MSa
✓ M9381A-M05	Memory, 512 MSa
M9381A-M10	Memory, 1024 MSa
Other	
M9381A-UNT	Analog Modulation
M9381A-UK6	Commercial calibration certificate with test data for M9381A (M9301A, M9310A, M9311A)
M9300A-UK6	Commercial calibration certificate with test data for M9300A (module only)
Related Products in Recommended Configuration	
✓ M9036A	PXIe Embedded Controller
✓ M9018A	18-Slot PXIe Chassis

Software Information

Supported operating systems	Microsoft Windows® XP (32-bit) Microsoft Windows® 7 (32/64-bit) Windows Vista®, SP1 and SP2 (32-bit and 64-bit)
Standard compliant drivers	IVI-COM, IVI-C, LabVIEW, MATLAB
Supported application development environments (ADE)	VisualStudio® (VB.NET, C#, C/C++), VEE, LabVIEW, LabWindows/CVI, MATLAB
Agilent IO Libraries (version 16.2 or newer)	Includes: VISA Libraries, Agilent Connection Expert, IO Monitor
Signal Studio Software Connectivity: • N76xxB-9TP connects to M9381A, transportable perpetual license. • N7650B-2xx provides 5/50 waveform pack licenses. (Playback on up to four channels per license)	N7600B W-CDMA/HSPA+ N7601B cdma2000/1xEV-DO N7602B GSM/EDGE/Evo N7606B Bluetooth N7611B Broadcast Radio N7612B TD-SCDMA/HSDPA N7615B Mobile WiMAX N7617B WLAN 802.11a/b/g/n/ac N7623B Digital Video N7624B LTE/LTE-Advanced FDD N7625B LTE/LTE-Advanced TDD
Agilent Command Expert	Instrument control for SCPI or IVI-COM drivers

Advantage Services: Calibration and Warranty

Agilent Advantage Services is committed to your success throughout your equipment's lifetime

R-51B-001-3C	Return to Agilent Warranty - 3 years
R-51B-001-5C	Return to Agilent Warranty - 5 years

The Modular Tangram

The four-sided geometric symbol that appears in this document is called a tangram. The goal of this seven-piece puzzle is to create identifiable shapes—from simple to complex. As with a tangram, the possibilities may seem infinite as you begin to create a new test system. With a set of clearly defined elements—hardware, software—Agilent can help you create the system you need, from simple to complex.

Challenge the Boundaries of Test

Agilent Modular Products

PXI www.pxisa.org

Agilent Solutions Partners

www.agilent.com/find/solutionspartners

Agilent
Advantage
Services

Agilent Advantage Services is committed to your success throughout your equipment's lifetime.

www.agilent.com/find/advantageservices

Agilent Email Updates

www.agilent.com/find/emailupdates

www.agilent.com
www.agilent.com/find/modular
www.agilent.com/find/m9381a

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**

Americas

Canada	(877) 894 4414
Brazil	(11) 4197 3600
Mexico	01800 5064 800
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
Other AP Countries	(65) 375 8100

Europe & Middle East

Belgium	32 (0) 2 404 93 40
Denmark	45 45 80 12 15
Finland	358 (0) 10 855 2100
France	0825 010 700*
	*0.125 €/minute
Germany	49 (0) 7031 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
United Kingdom	44 (0) 118 927 6201

For other unlisted Countries: **www.agilent.com/find/contactus**

Revised: January 6.2012

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

© Agilent Technologies, Inc. 2012
Printed in USA, August 29, 2012
5991-1050EN

Agilent Technologies