Oscilloscope Memory Architectures – Why All Acquisition Memory is Not Created Equal

Application Note

Introduction

Many people would say their car could never have too much gas mileage or their house could never be too large. On a similar note, many oscilloscope users would say their scope could never have too much acquisition memory. But, just like there are tradeoffs in gas mileage (less acceleration for example) or in the size of your house (more costs to heat/cool), depending on your oscilloscope architecture there may be very real tradeoffs in more acquisition memory.

In this application note, we will talk about:

- Why oscilloscope acquisition memory is important
- Different oscilloscope architectures and the benefits and drawbacks of those architectures
- Different techniques to best use your oscilloscope's acquisition memory

Why Oscilloscope Acquisition Memory is Important

Acquisition memory is an integral part of any oscilloscope. In an oscilloscope's simplest form, it is made up of a front end for acquiring the analog signal; that signal is then passed on to an analog to digital converter where the signal is digitized. Once it is digitized, that information has to be stored in memory (acquisition memory), processed and plotted/displayed. The oscilloscope acquisition memory is directly tied to the sample rate. The more memory you have, the higher you can keep the oscilloscope's sample rate as you capture longer periods of time. The higher the sample rate, the higher the effective bandwidth of the oscilloscope (up to the maximum bandwidth of the oscilloscope's front end).

So the deeper the memory, the better the oscilloscope, right? All things being equal, the answer would be yes. Let's compare two oscilloscopes with similar specifications outside of memory depth. One is a 1 GHz scope with 5GS/s sample rate and 4,000,000 points (4Mpts) of acquisition memory (we'll call this a "MegaZoom Architecture"). The other is a 1 GHz scope with 5GS/s sample rate and 20,000,000 points (20Mpts) of acquisition memory (we'll call this a "CPU-based Architecture"). Table 1 shows common time base settings along with the sample rate. There is a simple calculation to determine the sample rate given a specified time base setting and a specific amount of memory (assuming 10 divisions across screen and no off-screen memory captured):

Memory depth / ((time per division setting) * 10 divisions) = sample rate (up to the max sample rate of the ADCs)

For example, let's assume a time base setting of 160uS/div and a max memory depth of 4,000,000 samples. That would be 4,000,000 / ((160uS/div) * 10 divisions) = 2.5GS/s.

As table 1 shows, the deeper the memory, the higher the sample rate will be as you move in to slower time/div settings. Maintaining high sample rate is important as it allows the scope to function at its maximum capabilities. There is a wide range of memory depths available today in scopes with 5GS/s sample rates, from 10,000 points (10Kpts) all the way up to 1,000,000,000 points (1Gpts).

		4Mpts of Acquisition Memory	20Mpts of Acquisition Memory
400	pS/div	5GS/s	5GS/s
1	nS/div	5GS/s	5GS/s
2	nS/div	5GS/s	5GS/s
4	nS/div	5GS/s	5GS/s
10	nS/div	5GS/s	5GS/s
20	nS/div	5GS/s	5GS/s
40	nS/div	5GS/s	5GS/s
100	nS/div	5GS/s	5GS/s
200	nS/div	5GS/s	5GS/s
400	nS/div	5GS/s	5GS/s
1	uS/div	5GS/s	5GS/s
2	uS/div	5GS/s	5GS/s
4	uS/div	5GS/s	5GS/s
10	uS/div	5GS/s	5GS/s
20	uS/div	5GS/s	5GS/s
40	uS/div	5GS/s	5GS/s
100	uS/div	4GS/s	5GS/s
200	uS/div	2GS/s	5GS/s
400	uS/div	1GS/s	5GS/s
800	uS/div	500MS/s	2.5GS/s
2	mS/div	200MS/s	1GS/s
4	mS/div	100MS/s	500MS/s
8	mS/div	50MS/s	250MS/s
20	mS/div	20MS/s	100MS/s

Table 1: Sample rates for two identical scopes with different memory depths at common time per division settings.

Oscilloscope Architectures

Deep memory is clearly beneficial when it comes to sample rate, so when would it not be advantageous? It becomes a bad thing when it makes your oscilloscope so slow that it is no longer helpful in debugging a problem. Deep memory puts a larger strain on the system. Some scopes are set up to handle that well and remain responsive with a fast update rate; others attempt to make it a banner specification when it isn't really usable and slows the update rate by orders of magnitude.

Update rate (sometimes called "dead time") is how fast an oscilloscope can trigger, process the data it has captured, and then display it to the oscilloscope's screen. The faster the update rate, or the shorter the dead time, the more likely you are to catch an infrequent event. People often associate fast update rates with analog oscilloscopes from years ago. Fortunately, new oscilloscope architectures like Agilent's MegaZoom IV allow for even faster update rates than the fastest analog scopes of yesteryear.

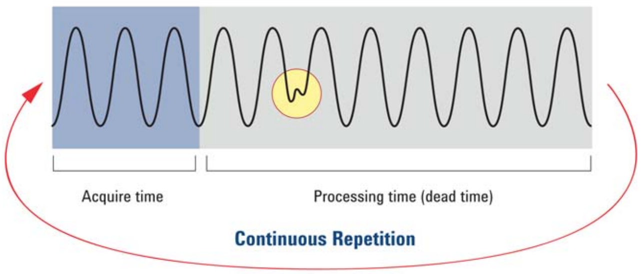


Figure 1: Oscilloscope dead time can hide rare events. A faster update rate (the inverse of dead time) can help increase your chances of seeing those infrequent events.

Let's look at those same two scopes from above. At 20nS/div (a fast time base setting), both scopes are near their maximums for update rate, and neither scope is using its full memory that is specified in their data sheet. But, what happens when you look at another time base setting like 400nS/div? The MegaZoom architecture oscilloscope automatically maximizes its memory depth to keep its sample rate maxed out. The scope will behave exactly as you would expect a deep memory scope to behave (it will keep its sample rate at 5GS/s and still have a fast update rate). The CPU-based architecture scope is still using its default memory depth to keep the scope responsive and isn't keeping its sample rate as high as it should (and still has a slower update rate). What happens if we adjust the memory depth to keep the sample rate high? You begin to see the trade-offs of a deep memory scope that isn't designed to handle deep memory; the user has to intervene and set the memory depth higher, which brings the sample rate up to its maximum (5GS/s), but the update rate is 1/6 the amount of the MegaZoom scope. It only gets worse as you look at slower time base settings (e.g. at 4uS/div the MegaZoom scope has an update rate almost 20 times faster than the CPU-based scope).

		MegaZoom Architecture			CPU-Based Architecture		
Time base setting	**MSO Enabled**	**Sample Rate**	**Update Rate**	**Memory Depth**	**Sample Rate**	**Update Rate**	**Memory Depth**
10nS/Div	No	5GS/s	1,090,000wfms/s	Auto-adjust	5GS/s	3,000wfms/s	10Kpts
20nS/Div	No	5GS/s	840,000wfms/s	Auto-adjust	5GS/s	64,000wfms/s	10Kpts
100nS/Div	Yes	5GS/s	238,000wfm/s	Auto-adjust	5GS/s	120wfms/s	10Kpts
400nS/Div	No	5GS/s	74,000wfms/s	Auto-adjust	2.5GS/s	57,000wfms/s	10Kpts
400nS/Div	No	5GS/s	74,000wfms/s	Auto-adjust	5GS/s	12,400wfms/s	100Kpts
4uS/Div	No	5GS/s	7,800wfms/s	Auto-adjust	5GS/s	400wfms/s	1Mpts

Table 2: Comparison of update rates, sample rates and memory depths.

What makes one scope "designed" for deep memory while another has to default its memory to 10K to remain responsive? A lot of it comes down to the oscilloscope architecture. In some scopes, the CPU system is an integral block in the oscilloscope architecture ("CPU-based architecture"), so much so that it is actually the gating item in how fast the scope can process the information and display it to the screen If the CPU system isn't up to the task of handling deep memory acquisition records, it will lengthen the time it takes to process and display the data, therefore lowering the update rate of the scope (sometimes dramatically). See Figure 2 for an example of this architecture.

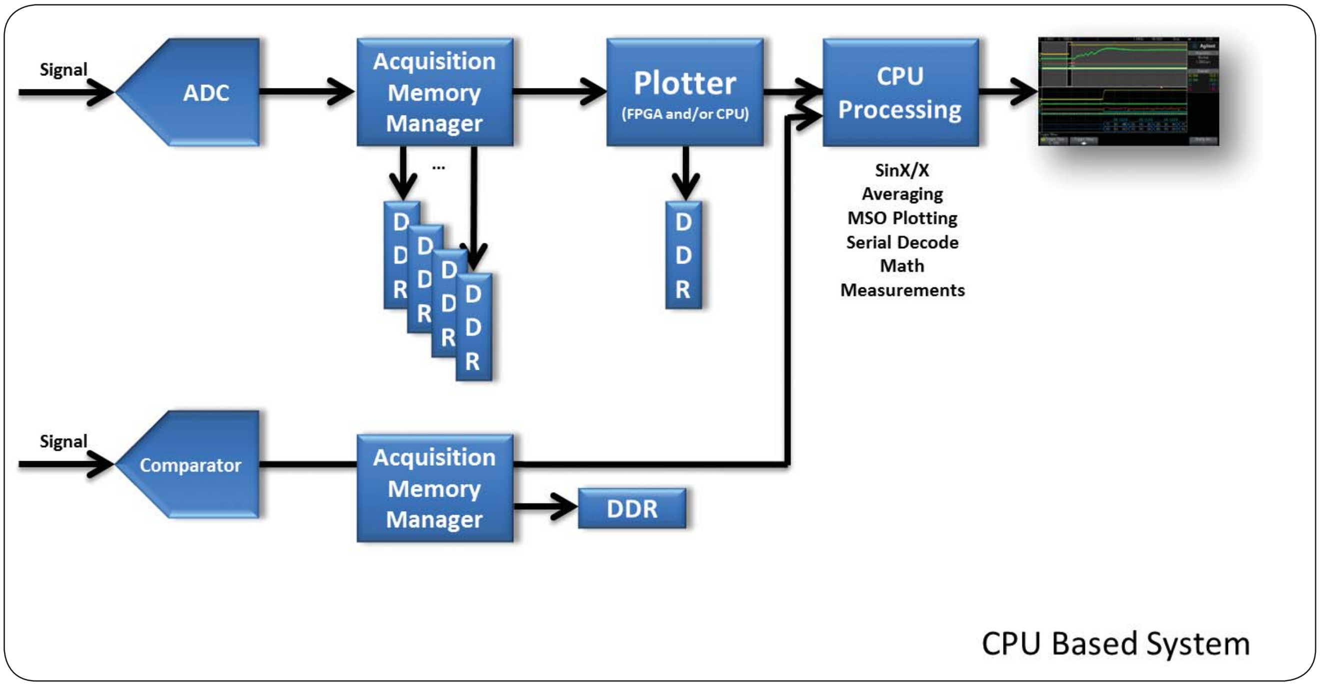


Figure 2: CPU-based Architecture Block Diagram showing how the CPU system is a bottle neck to overall waveform plotting.

Fortunately, there is another way. In the scope designed for deep memory, it uses a custom ASIC that eliminates the need for the oscilloscope's CPU to be an integral part of the architecture. Is there still a CPU system? Of course, but it is now used for peripheral crunching of data, which allows the scope to focus on what it does best: displaying waveforms. Figure 3 shows an example of this innovative architecture in the DSO-X 4000 Series from Agilent, which uses a custom ASIC (called MegaZoom IV) to provide fast update rates while maximizing memory and sample rate.

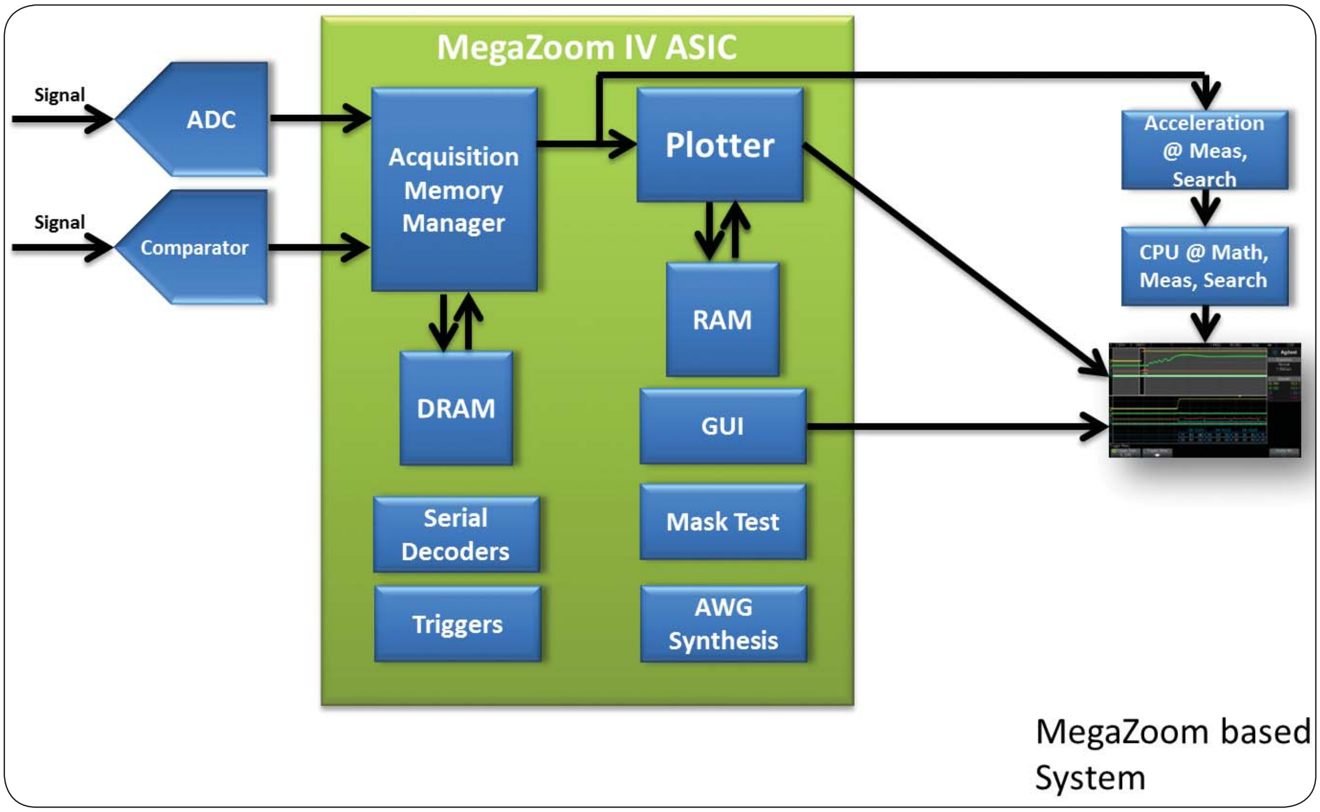


Figure 3: MegaZoom architecture with a custom ASIC driving the plotting of waveforms from acquisition memory.

Memory and the oscilloscope's architecture are so intertwined that there are some things that even defaulting to a base memory depth of 10K can't fix. For example, one of the best enhancements of scopes in the last 15 years is the addition of digital channels, but not all digital channels are implemented the same. In the CPU-based architecture that we discussed above, turning on the digital channels will actually cause such a slowdown in the oscilloscope that the update rate will never get above 125 waveforms per second regardless of the memory depth or time base setting (see Table 2 at 100nS/div for an example). That is orders of magnitude slower than the maximum update rate the manufacturer specifies. Why is that? Again, it goes back to the oscilloscope architecture. As you can see in Figure 2, the MSO channels are not well integrated into the CPU-based architecture, which requires the CPU system to have a larger role in plotting them. With the MegaZoom architecture (Figure 3), you can see the digital channels are an integral part of the custom ASIC that is doing the plotting and displaying of all the channels. In the MegaZoom architecture, you will not see a significant slowdown due to turning on digital channels. Other common things like Sinx/x interpolation can also slow down a CPU-based system – so much so that you'll see dramatic drop offs in update rate when moving between time base settings as the scope switches on and off Sinx/x interpolation (see Table 2 at 10nS/div for an example). The MegaZoom architecture does not suffer from this issue.

Responsiveness of the scope is another drawback to a CPU-based system. Have you ever changed the time base setting on your deep memory scope and then waited for it to catch up? Or, have you tried to update a setting only to have it respond slowly and you accidentally click past the setting you were trying to get? That is because of the CPU system trying to crunch through the data – the same issue that causes the update rate to slow is also causing the scope's responsiveness to slow.

How to Best Use your Oscilloscope's Memory

So far all we've discussed are modes when the oscilloscope is running and being used for something like debug. If you are just looking at a single shot acquisition, deep memory is better again right? You don't need a fast update rate with a single shot acquisition, and the responsiveness of the scope should be better once it is all captured and displayed. Again, this would seem like a logical conclusion, and in some cases this is true, but what if you are looking at a signal that has bursts of information with a significant amount of idle time in between (like a radar pulse or a serial bus sending frames/packets)? With a traditional deep memory oscilloscope, you would be using all the memory to digitize the idle time and the bursts – this is not the best use of the memory since you probably only care about the signal bursts themselves. Some oscilloscopes offer a memory system that has a capability called "segmented" memory. Segmented memory allows you to digitize just the portion of the waveform that you care about so you can make more efficient use of your deep memory.

Let's look at an example where segmented memory might be advantageous. In Figure 4 you can see two RF pulses separated by a long period of idle time in between. In a traditional deep memory oscilloscope, we are digitizing the bursts and the idle time. As you can see in Figure 4, the sample rate for the scope (which typically samples at 5GS/s) is only 313MSa/s – and that is with us just capturing two of the pulses! What would happen if we wanted to capture 250 of the pulses? The sample rate would drop to less than 10MS/s and the pulses would no longer be identifiable because they are severely under sampled. If we wanted to

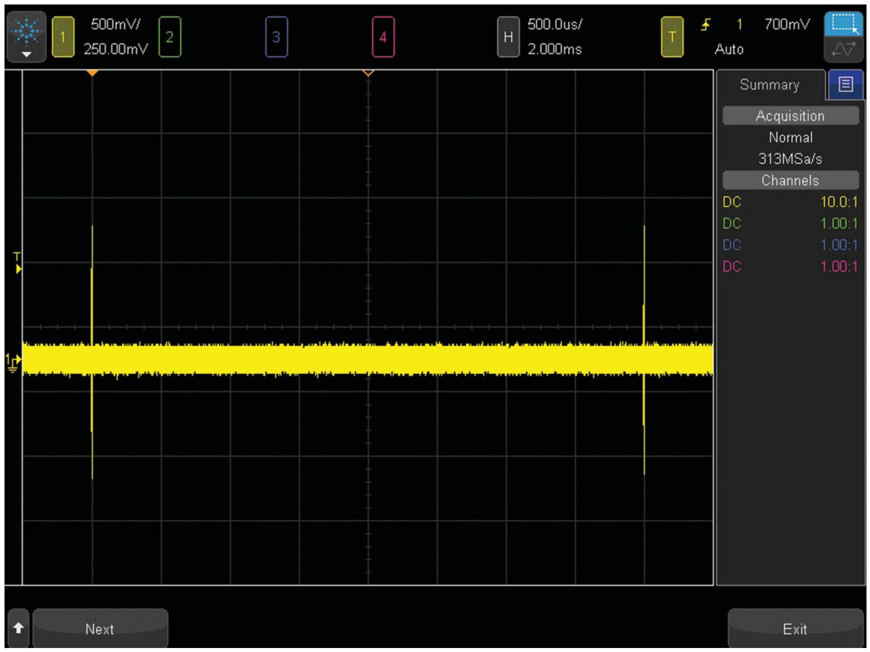


Figure 4: Two RF pulses spread out in time. Notice the lower sample rate due to the oscilloscope digitizing the pulses and the idle time between them.

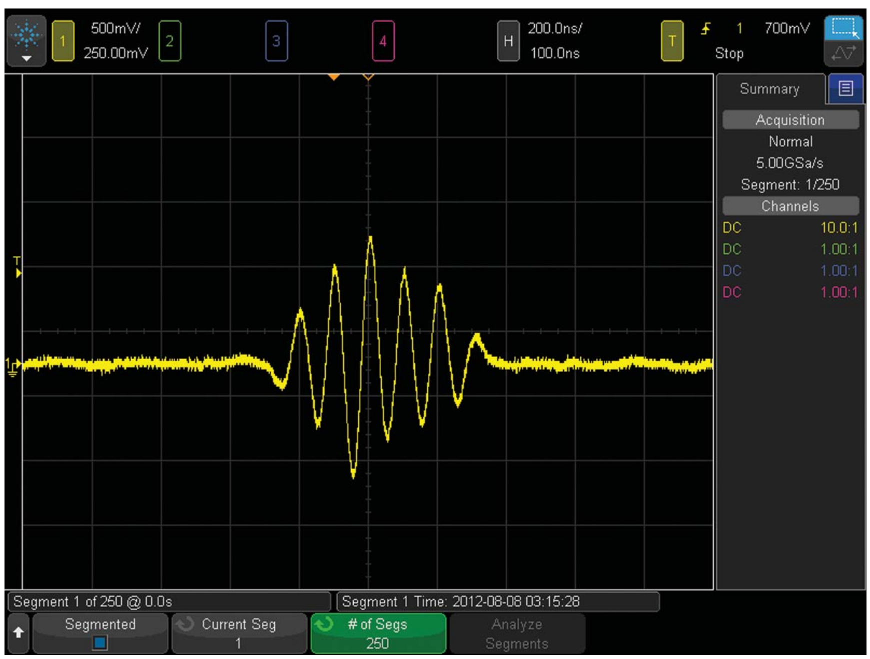


Figure 5: First RF pulse (1 of 250) captured using segmented memory – note the higher sample rate (5GS/s) versus the traditional method in Figure 4 of capturing just TWO pulses (313MS/s).

capture those 250 pulses and all the idle time between them at a sample rate of 5 GS/s, we would need an oscilloscope with 5.0 gigapoints of memory (5,000,000,000). No scope on the market today offers that deep of memory. With segmented memory, we are able to digitize just the portion of the waveform that we care about (the burst itself) and ignore all of the idle time in between bursts. Figure 5 shows the first RF pulse captured using segmented memory – note the sample rate was 5GS/s and each segment was time stamped so you know exactly when it happened in relation to the initial trigger. Figure 6 shows the 250th pulse and its time stamp (996.004ms). The oscilloscope allows you to walk through each of the segments and analyze them (including decoding of each segment's packets/frames if you were using segmented memory with a serial bus).

By combining the MegaZoom architecture with the intelligence of segmented memory, the user can not only get the benefits of a fast, responsive oscilloscope, but also the time capture of an ultra-deep memory oscilloscope as well.

Summary

While a data sheet with a large number for acquisition memory can be tempting, you should definitely consider how you will be using the scope. In some cases, the deepest memory possible will be the best option, but in many cases, a scope designed to handle deep memory is going to be a better option and create less frustration from sluggishness or odd operating modes. In addition, some new advancements in efficiently utilizing the oscilloscope's memory can greatly enhance your oscilloscope's capabilities.

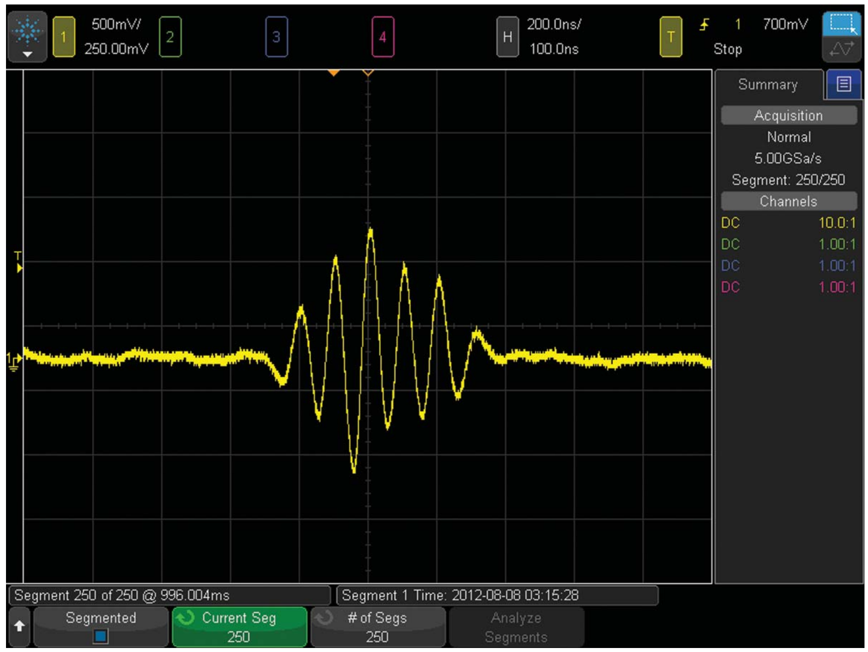


Figure 6: Last RF pulse (#250) captured using segmented memory – note that it maintained the 5GS/s sample rate and the elapsed time of almost 1 second.

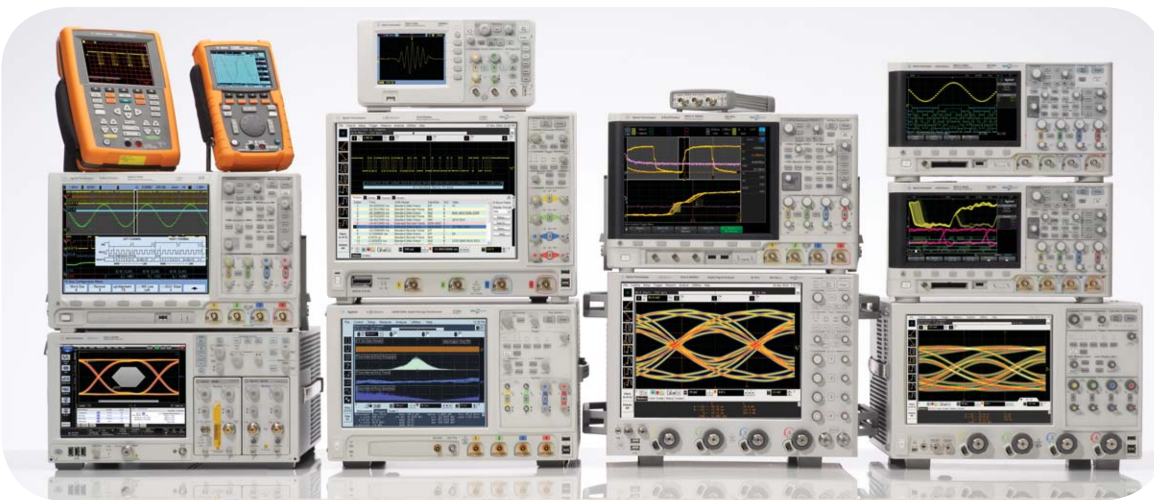
Related Literature

Publication Title	Publication Type	Publication Number
Agilent InfiniiVision 4000 X-Series Oscilloscopes	Data sheet	5990-1103EN
InfiniiVision Oscilloscope Probes & Accessories	Data sheet	5968-8153EN
Evaluating Oscilloscopes for Best Waveform Update Rates	Application Note	5989-7885EN
Oscilloscope Display Quality Impacts Ability to View Subtle Signal Details	Application Note	5989-2003EN

To download these documents, insert the publication number in the URL: http://cp.literature.agilent.com/litweb/pdf/xxxx-xxxxEN.pdf

Product Web site

For the most up-to-date and complete application and product information, please visit our product Web site at: www.agilent.com/find/morescope



Agilent Technologies Oscilloscopes

Multiple form factors from 20 MHz to >90 GHz | Industry leading specs | Powerful applications

www.agilent.com
www.agilent.com/find/morescope

myAgilent

www.agilent.com/find/myagilent
A personalized view into the information most relevant to you.

www.axiestandard.org
AdvancedTCA® Extensions for Instrumentation and Test (AXIe) is an open standard that extends the AdvancedTCA for general purpose and semiconductor test. Agilent is a founding member of the AXIe consortium.

www.lxistandard.org
LAN eXtensions for Instruments puts the power of Ethernet and the Web inside your test systems. Agilent is a founding member of the LXI consortium.

www.pxisa.org
PCI eXtensions for Instrumentation (PXI) modular instrumentation delivers a rugged, PC-based high-performance measurement and automation system.

Agilent Channel Partners

www.agilent.com/find/channelpartners
Get the best of both worlds: Agilent's measurement expertise and product breadth, combined with channel partner convenience.

Agilent Advantage Services

Agilent Advantage Services is committed to your success throughout your equipment's lifetime. We share measurement and service expertise to help you create the products that change our world. To keep you competitive, we continually invest in tools and processes that speed up calibration and repair, reduce your cost of ownership, and move us ahead of your development curve.

www.agilent.com/find/advantageservices

Agilent Electronic Measurement Group
KEMA Certified
ISO 9001:2008
Quality Management System
www.agilent.com/quality

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: October 11, 2012

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

Published in USA, October 26, 2012
5991-1024EN

Agilent Technologies