MSO

Digital Channels

All contemporary Siglent DSOs can be upgraded to MSOs. The cheaper low-end devices like the SDS800/1000X HD don't have the required hardware built in, but communicate with a completely autonomous subsystem: it's the SLA1016, which has already been introduced back in early 2018 to add MSO capabilities to the SDS1004X-E and later 2000X-E series.

My review from that time shows the hardware, which is nowhere near as sexy as the fully integrated SPL2016 solution:

The first implementation of the MSO option has become available with firmware 7.6.1.20 for the SDS1004X-E and I've received the SLA1016 digital probe about a week ago. Unfortunately, the Sbus-cable was missing and even though the connectors are identical, an HDMI cable cannot replace it. So I have to wait until I get the original one.

Until then, I'll show some details of the hardware. First the contents of the box (minus the Sbus cable):

Fig. 149 SLA1016_Set 01

Top left is the interface box SLA1016 that connects to the SDS1004X-E through the Sbus cable at one end. The other end has a 2x34 connector for the ~80cm long high-density flat ribbon cable, which in turn connects to the SPL1016 probe head. It is connected in the picture above, but can easily be detached.

The opposite side of the probe head has two standard 1/10" 2x8 pin male connectors where the supplied probe leads can be connected as well as user specific probes, e.g. with a 16-wire flat ribbon cable on a standard 1/10" 2x8 female header.

The two supplied probe leads consist of a 1/10" 2x8 female header with eight 140mm long digital input leads and two 100mm long ground wires. At the end of each wire there is a metal sleeve that connects to 0.64mm pins on hooks or any other test points/connectors.

Finally, there is a bag with 20 cheap hooks.

The picture below shows the SPL1016 probe head. It has the relevant specifications printed on it as well as the connector layout and the color scheme for the supplied probe leads.

Fig. 150 SPL1016_Head 01

The next image shows the probe connector side of the SPL1016 head together with one probe lead assembly and a hook.

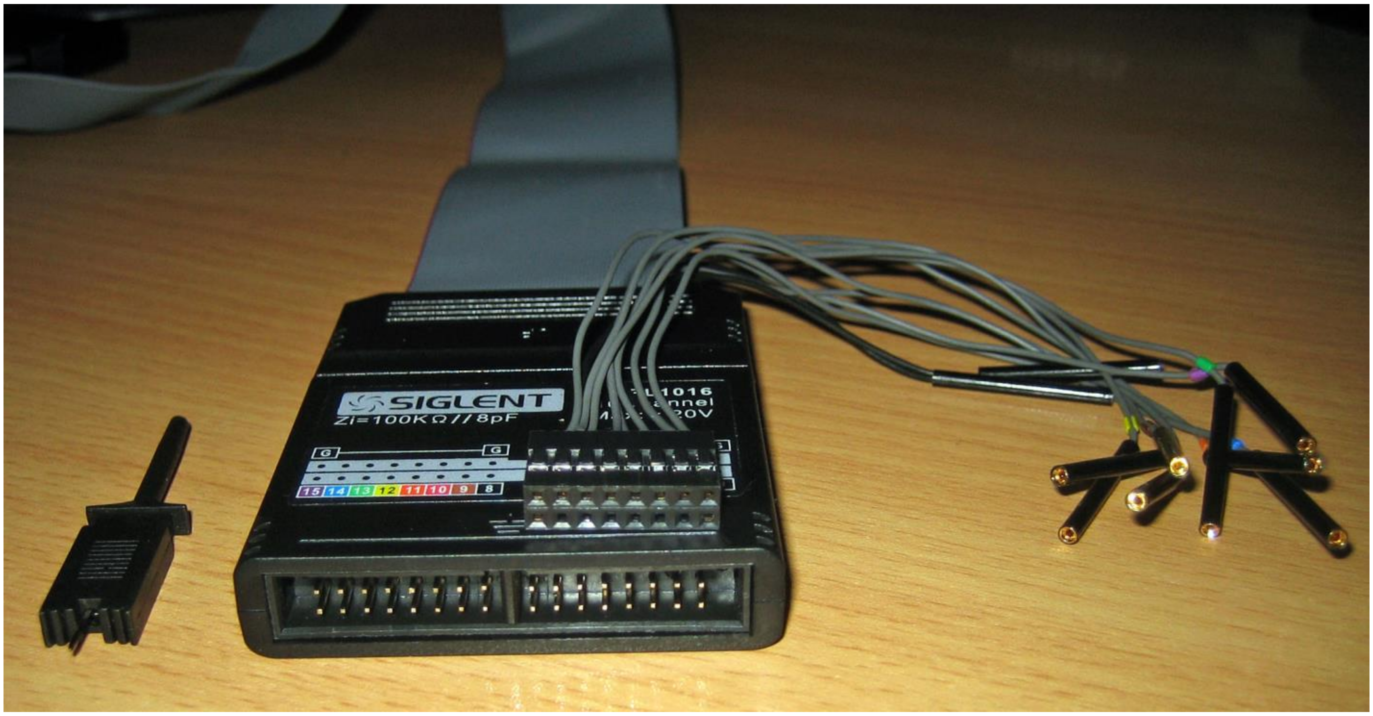

Fig. 151 SPL1016_Head 02

The tip of one of the hooks supplied with the SPL1016 is shown below. The diameter of the plastic sleeve is 1/10".

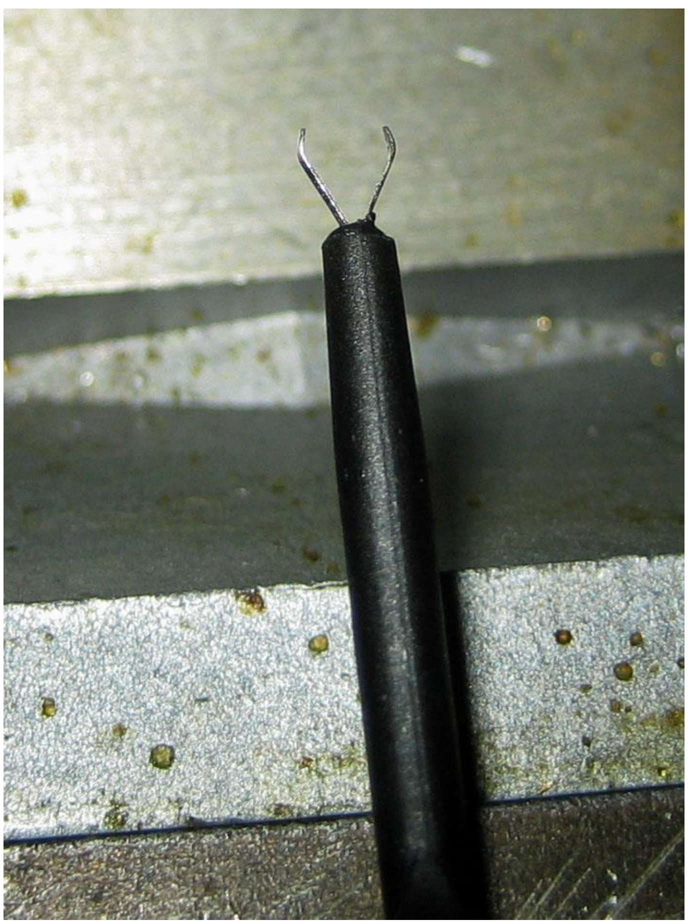

Fig. 152 SLA1016_Hook

Compare this with the E-Z hook (made in USA) supplied with the SPL2016 digital probe head that is an option for the SDS2000X oscilloscopes. The quality is appreciably better and the plastic sleeve has only 1/20" diameter at the tip.

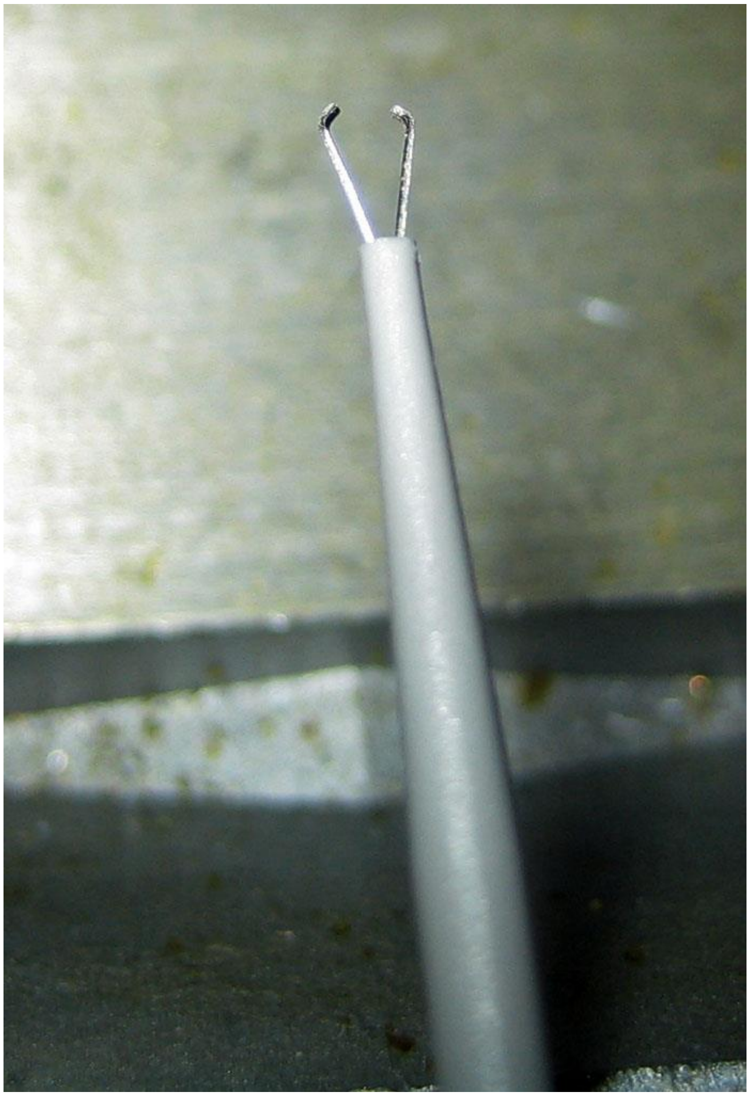

Fig. 153 SPL2016_E-Z-Hook

As a consequence, the no name hooks supplied with the SPL1016 can only connect every other pin on a SOIC with 1/20" pin spacing, whereas the SPL2016 hooks can connect eight adjacent SOIC pins in a row if required.

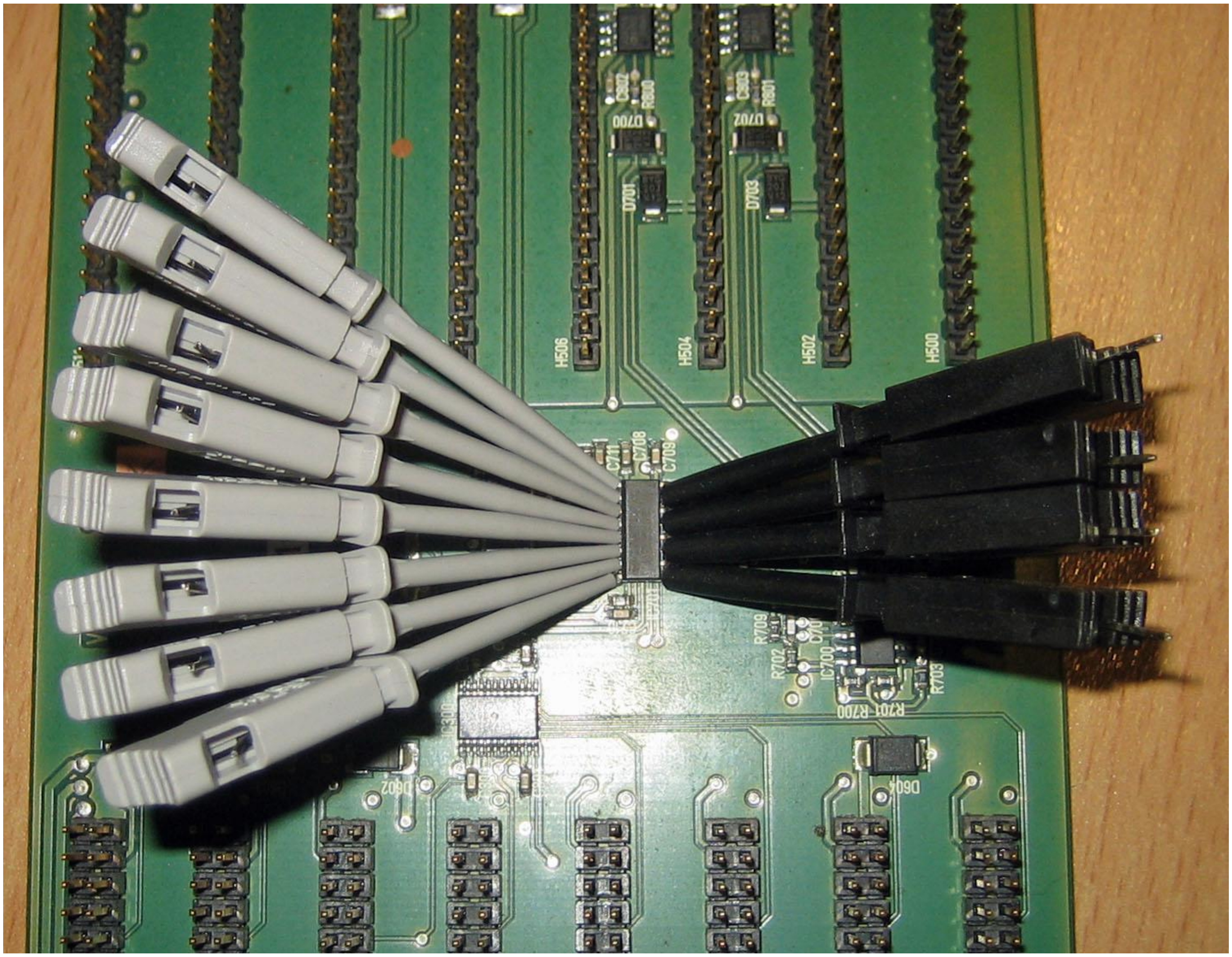

Fig. 154 Digital_Hooks_Spacing

Thankfully this is not an ultimate limitation for the SPL1016, because the E-Z Hooks (and probably most others) can be used together with the SPL1016 without any problems as shown below.

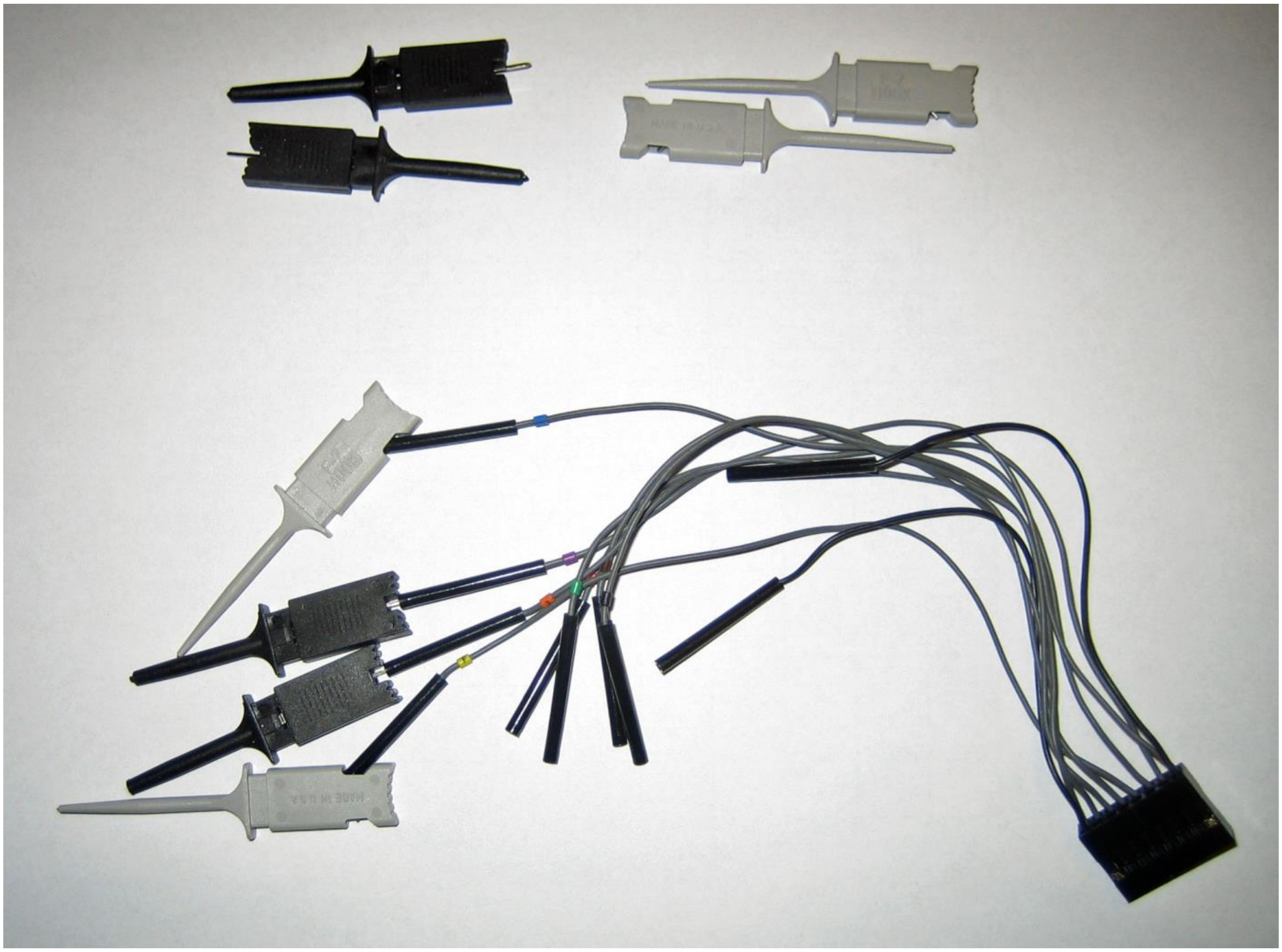
Fig. 155 Digital_Hooks 01

In order to work with the SDS800/1000X HD, the firmware in the SLA1016 needs to be upgraded. Unfortunately, the current version 8.2.3 provides no backward compatibility; once upgraded, the SLA1016 no longer works with SDS1004/2000X-E. If a single SLA1016 shall be shared between older SDS1004/2000X-E and new SDS800/1000X HD, the FW would have to be up- and downgraded accordingly.

The external subsystem approach has a few disadvantages:

- Since it incorporates a complete SOC and local memory, it cannot be cheap even though the probe head and the grabbers leave a cheap impression.
- Mixed analog / digital pattern trigger is not possible.
- Zoom mode cannot be used as soon as digital channels are enabled.
- History doesn't work either when digital channels are activated.

If you wonder where these limitations come from, it's simply because this is a separate subsystem connected via a (probably only moderately fast) serial interface. Because of the long memory, the DSO cannot have instant access to the full sample data through the SBUS interface (about 200 Mbit/s transfer speed needed for even only a single frame per second). Consequently, the SLA1016 only transfers the decimated screen data during normal operation.

I've already demonstrated its performance once in late 2018:

The integration of the MSO option with the SDS1004X-E DSO is not great, so you can call it crippled indeed, when compared to the fully integrated solutions in SDS1000X+ and SDS2000X.

Last time I've checked it nearly a year ago, there has been some 7ns skew between analog and digital channels and the digital de-skew parameter was not preserved upon a power cycle. I expect this bug to be fixed by now though.

Mixed channel Pattern Trigger is not supported, so it has to be either an analog or digital pattern.

Furthermore, both History and Zoom don't work when digital channels are enabled. Since most people want the MSO for decoding serial buses nowadays, this means you cannot decode long messages, because the decoder line at the bottom of the screen will become unreadable due to the lack of space. Using analog channels, you can easily deal with that by just entering zoom mode and take a closer look at the part of the message you're interested in. As it is now, when using digital channels, you're almost limited to the capabilities of a Rigol 1000Z which only decodes the screen buffer anyway

The limitation described above only applies in run mode though. You can still capture a long message and then zoom in and navigate through the message while in stop mode. Also, the list view works during Run even for very long messages, but that's rather pointless as you cannot closely examine the corresponding waveform – you might just as well use an LA instead.

Other than that, it's not all that bad. Here is an example how a 100MHz SDS1104X-E is capable of capturing a 320MHz sinewave with only 1.5Vpp amplitude on a logic channel of the SLA1016:

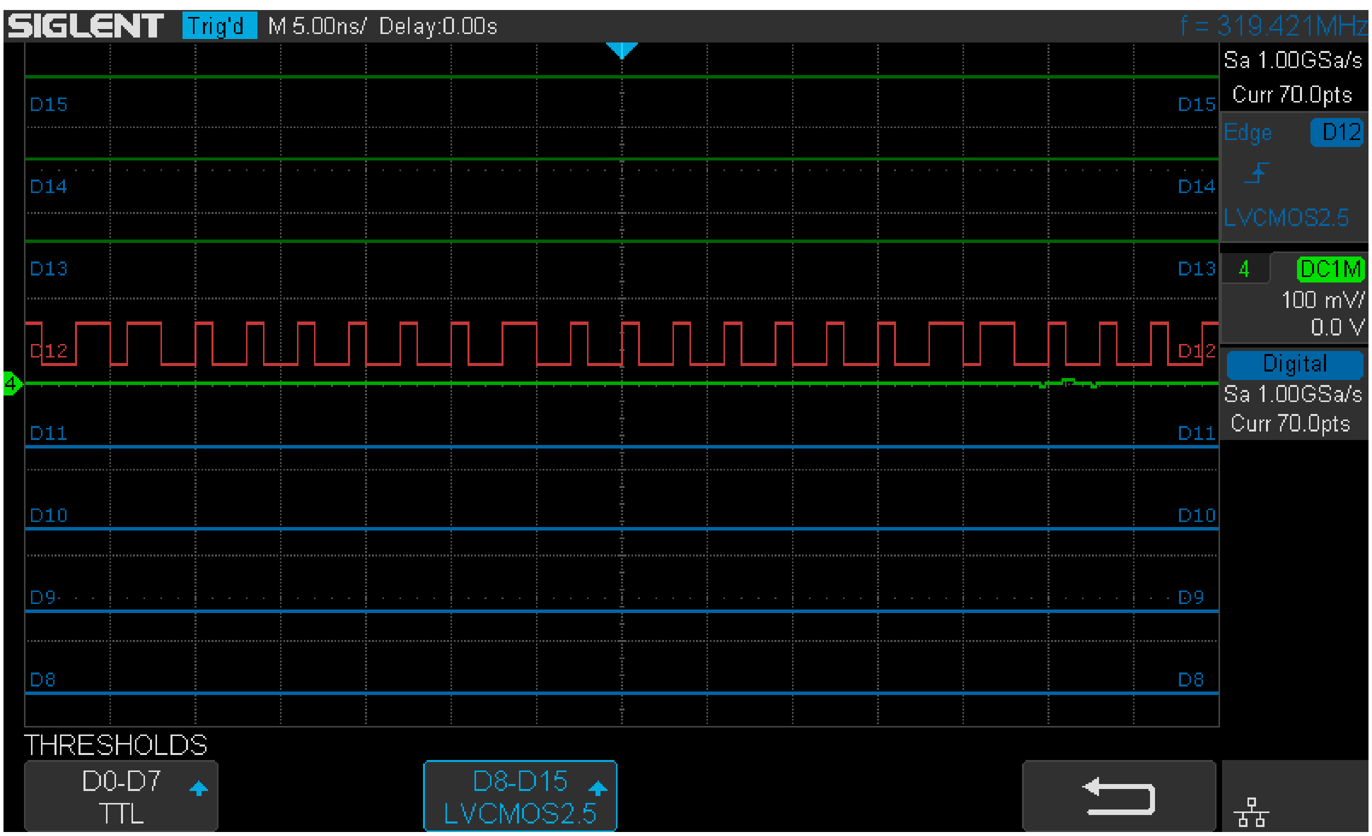

Fig. 156 SLA1016_D12_Sine_320MHz_1.5Vpp

A 4ns wide pulse – heavily distorted when viewed on the analog channel at just 100MHz bandwidth, but nicely captured on the digital channel:

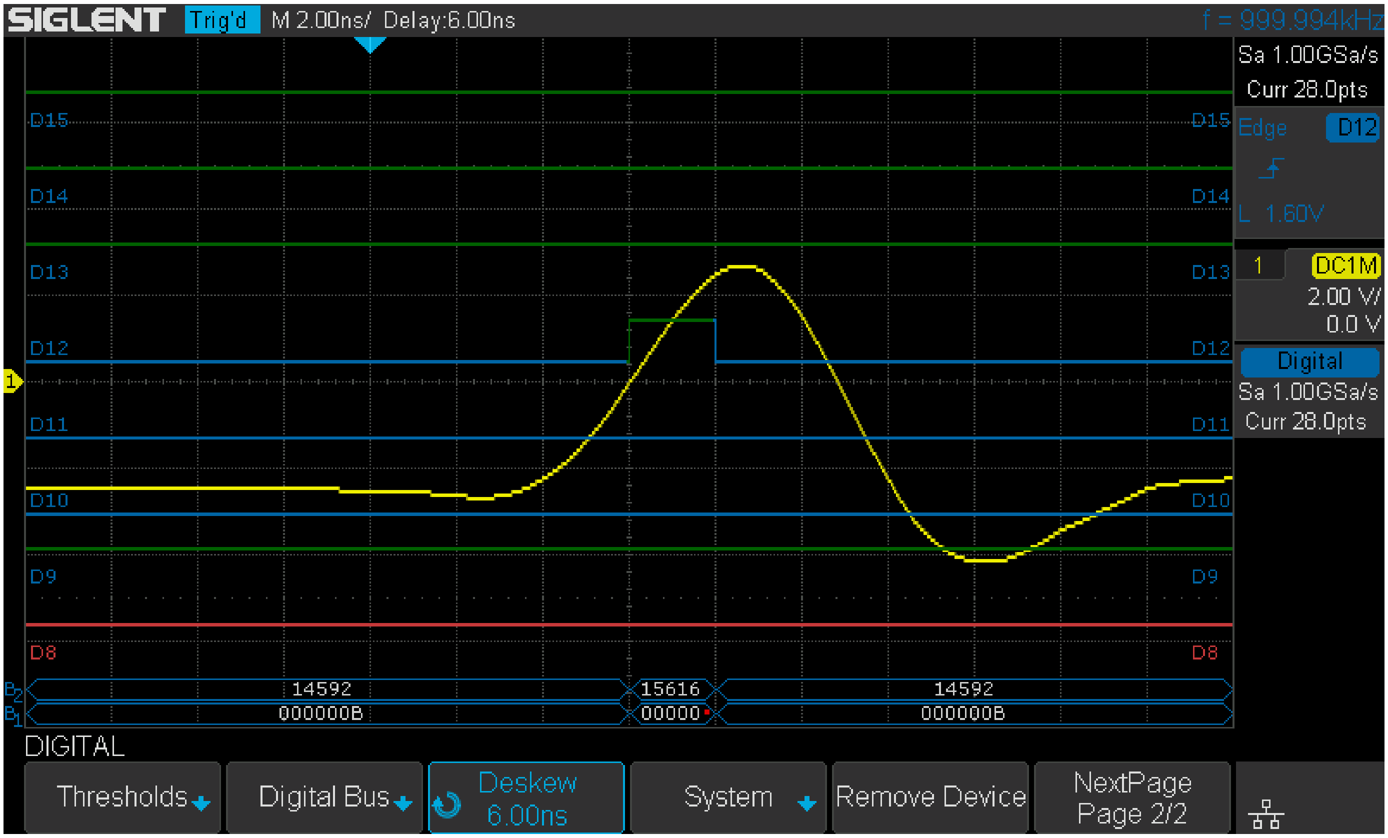

Fig. 157 SLA1016_AD_Pulse_5V_4ns_probe

16 digital channels, grouped into 2 parallel buses, captured at 1GSa/s with 14Mpts record length. This also demonstrates how the decoded values become unreadable in the busy regions, where we get just blue bars. In stop mode, zooming into these regions wouldn't be a problem though. I just don't happen to have a screenshot at hand for that.

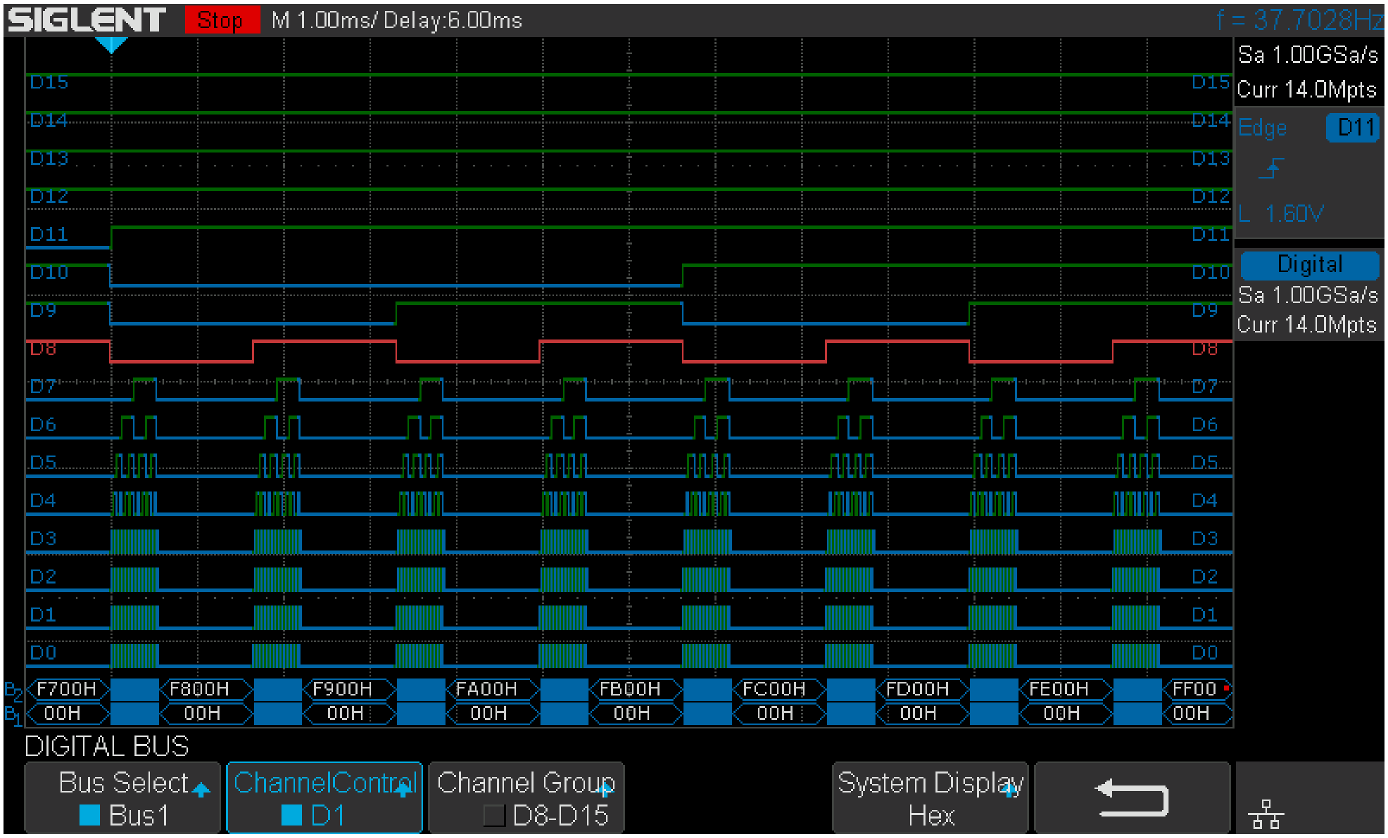

Fig. 158 SLA1016_Bus_Select

Finally, a digital pattern trigger. Two buses again, the first (lower) one decoded binary, the 2nd (upper) one decimal:

Fig. 159 SLA1016_Trig_Pattern_1us

Now let's add some more content. First, it might be important to know that the SDS800X HD doesn't collapse when its advertised capabilities are actually called up.

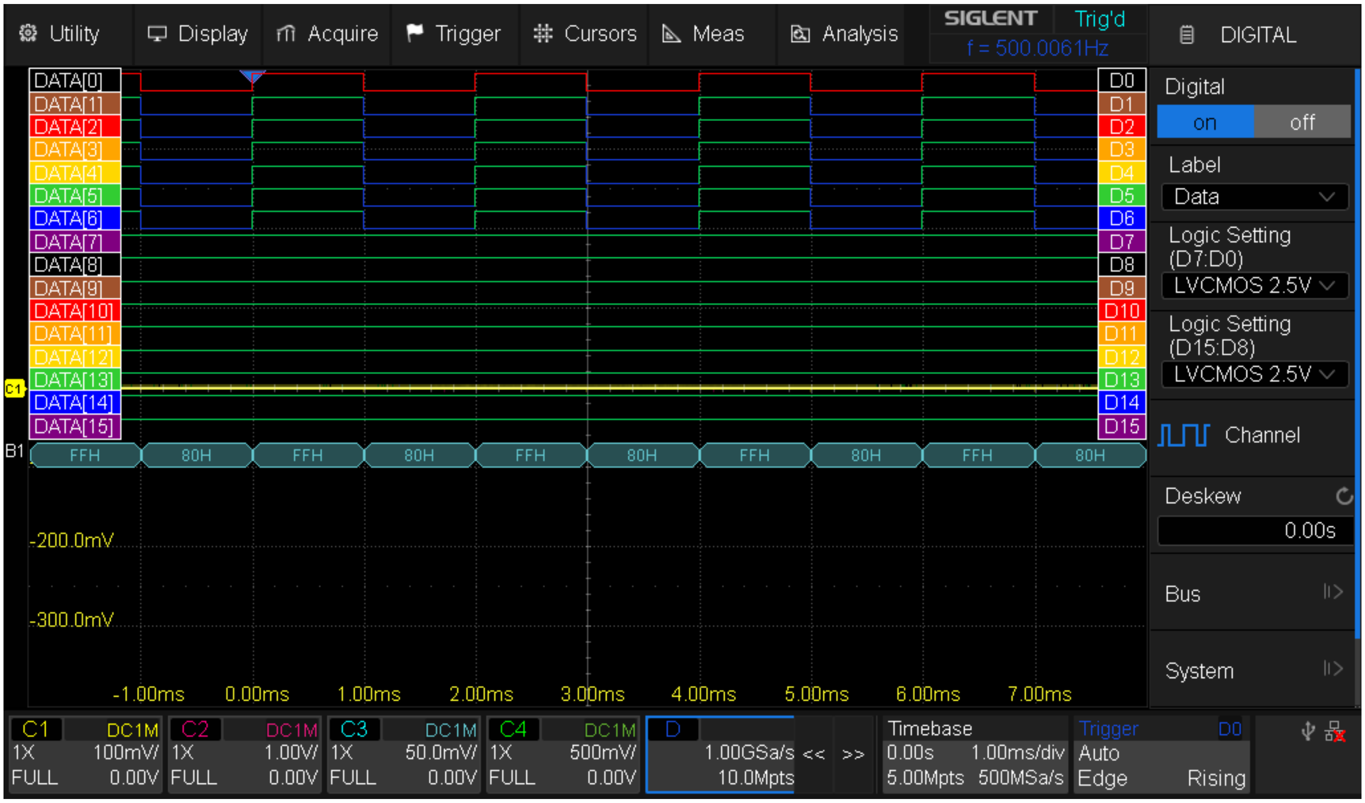

Fig. 160 SDS824X_HD_Digital_4Ch_P16_Deskew

In the above screenshot we can see 16 digital channels at 1 GSa/s sample rate together with 4 active analog channels at 500 MSa/s each.

We can further see one digital parallel bus decoder placed right under the digital traces, showing hexadecimal values.

The "Deskew" parameter is there to compensate for runtime differences between analog and digital probes; it can be adjusted in 10 ps steps.

There is a digital Edge trigger set on channel D0.

We can enable just 8 bits if we want to show e.g. the activity of a 7-bit counter:

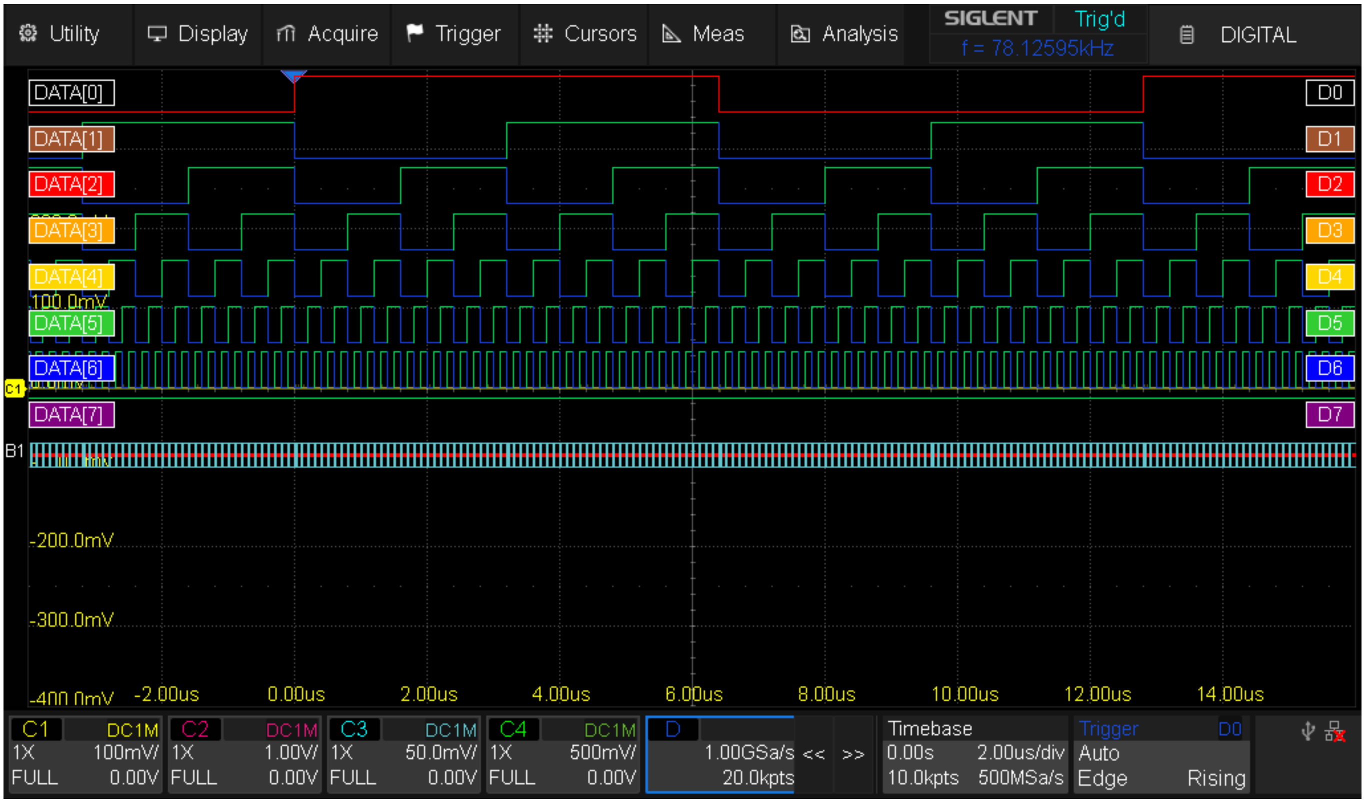

Fig. 161 SDS824X_HD_Digital_4Ch_P8_Counter

Now let's check a fast clock signal (on 7 channels):

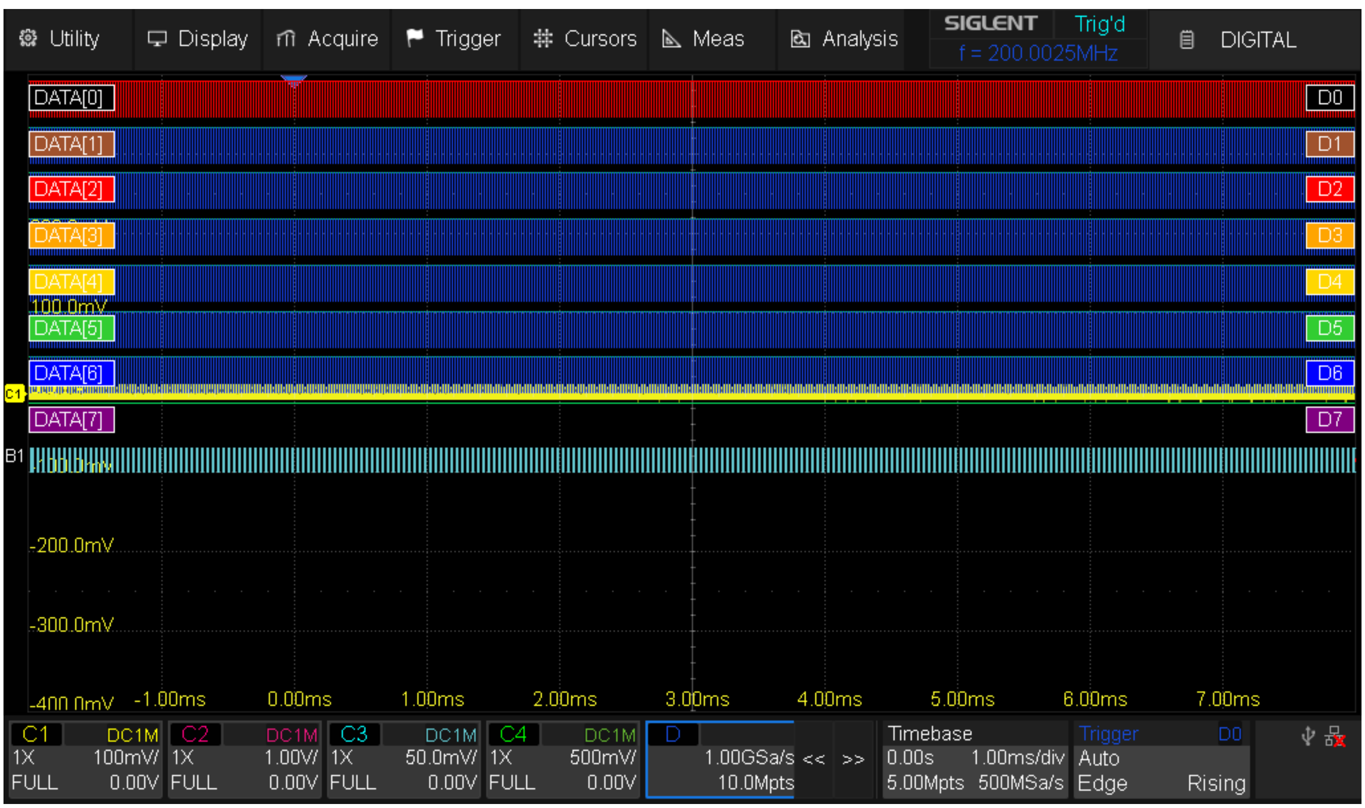

Fig. 162 SDS824X_HD_Digital_4Ch_P8_200MHz_1ms

In order to see the details. We need to stop the acquisition and then zoom in using a faster time base:

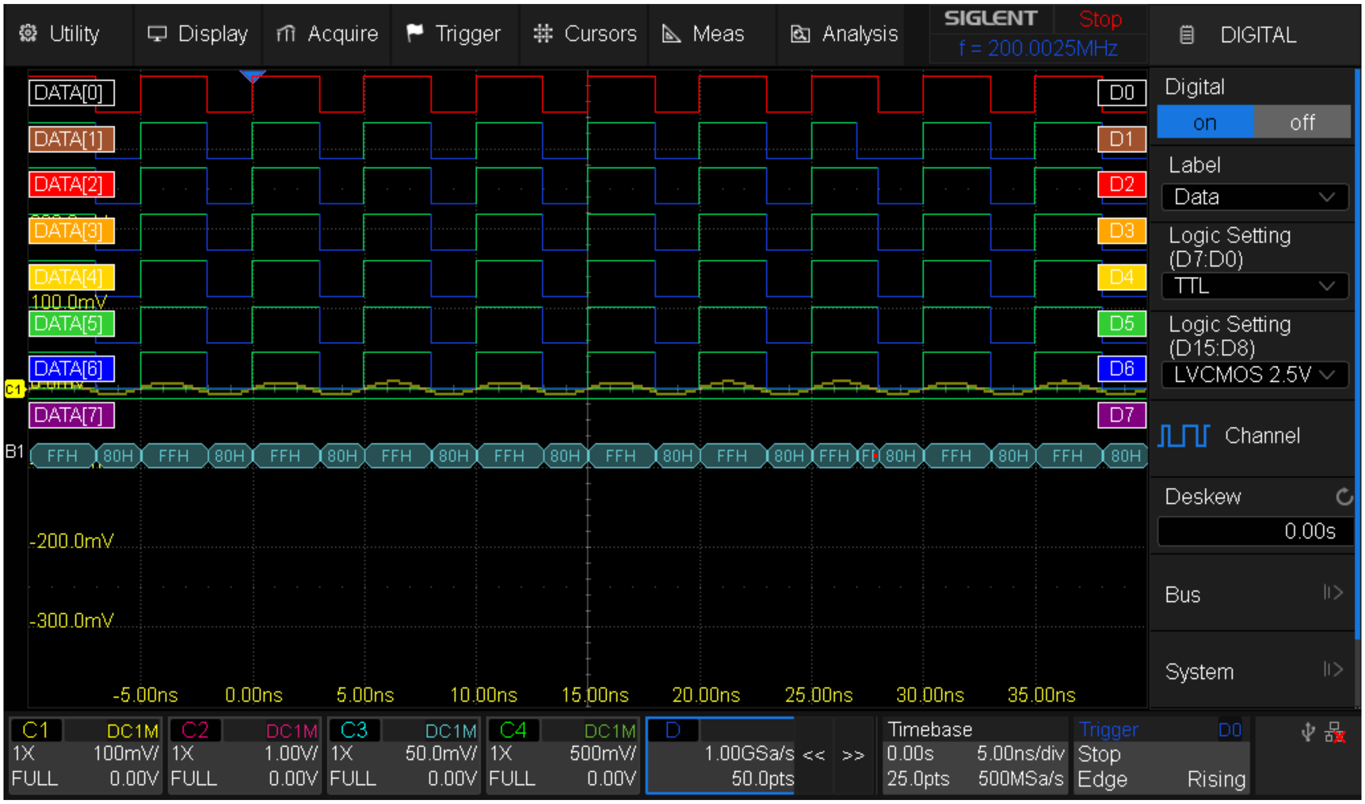

Fig. 163 SDS824X_HD_Digital_4Ch_P8_200MHz_Stop_Zoom_5ns

Who said we cannot have a perfect 200 MHz square wave on a 200 MHz bandwidth oscilloscope 😉

Here's another challenge: a 2 ns wide pulse with 500 ps rise time, captured analog and digital at the same time:

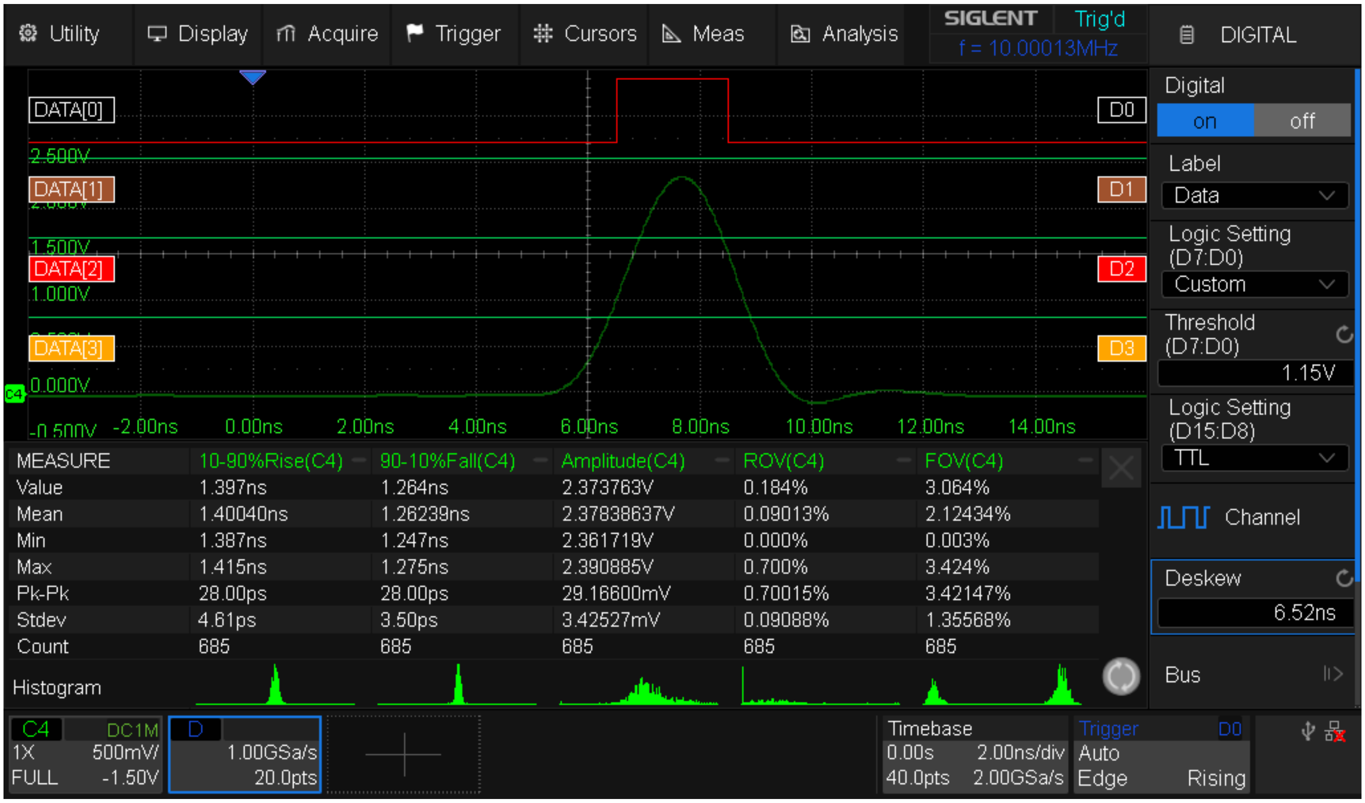

Fig. 164 SDS824X_HD_Digital_1Ch_p4_Pulse_2ns

A digital Deskew value of 6.52 ns was required to get both domains reasonably aligned; as expected, the digital channel shows a simplified representation of the pulse, which can be characterized by the measurements of the analog channel. Of course, 200 (or even 245) MHz bandwidth is not nearly enough to even remotely reproduce such a pulse; for this, at least 1 GHz bandwidth would be required. Yet it looks clean and good within the capabilities of a low bandwidth DSO and the transition time measurement results of 1.4 and 1.25 ns are encouraging too.

Probably the most popular use case nowadays would be serial decoding. I'll just demonstrate I2C, even though this protocol only requires two channels.

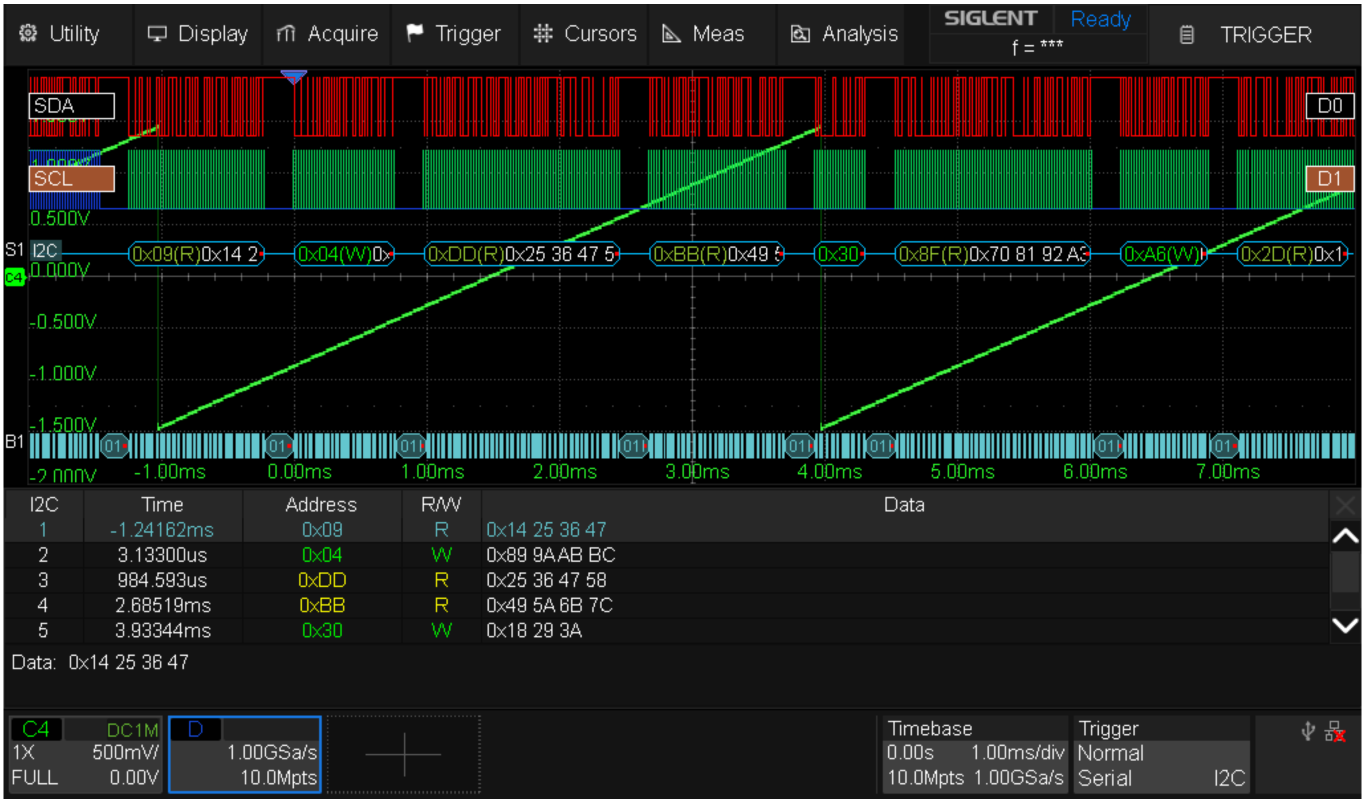

Fig. 165 SDS824X_HD_Digital_1Ch_I2C_Run

We can see some I2C messages in the decoder table at the bottom and two digital channels together with an analog trace in the upper half of the screen. Right in the middle, there is still the parallel bus decoder line, which is of course unreadable at this time base. The I2C decoder line right under the digital channels is readable, yet messages are truncated, as indicated by the red dots.

In stop mode, we can choose a faster time base and show the complete I2C messages in the decoder line.

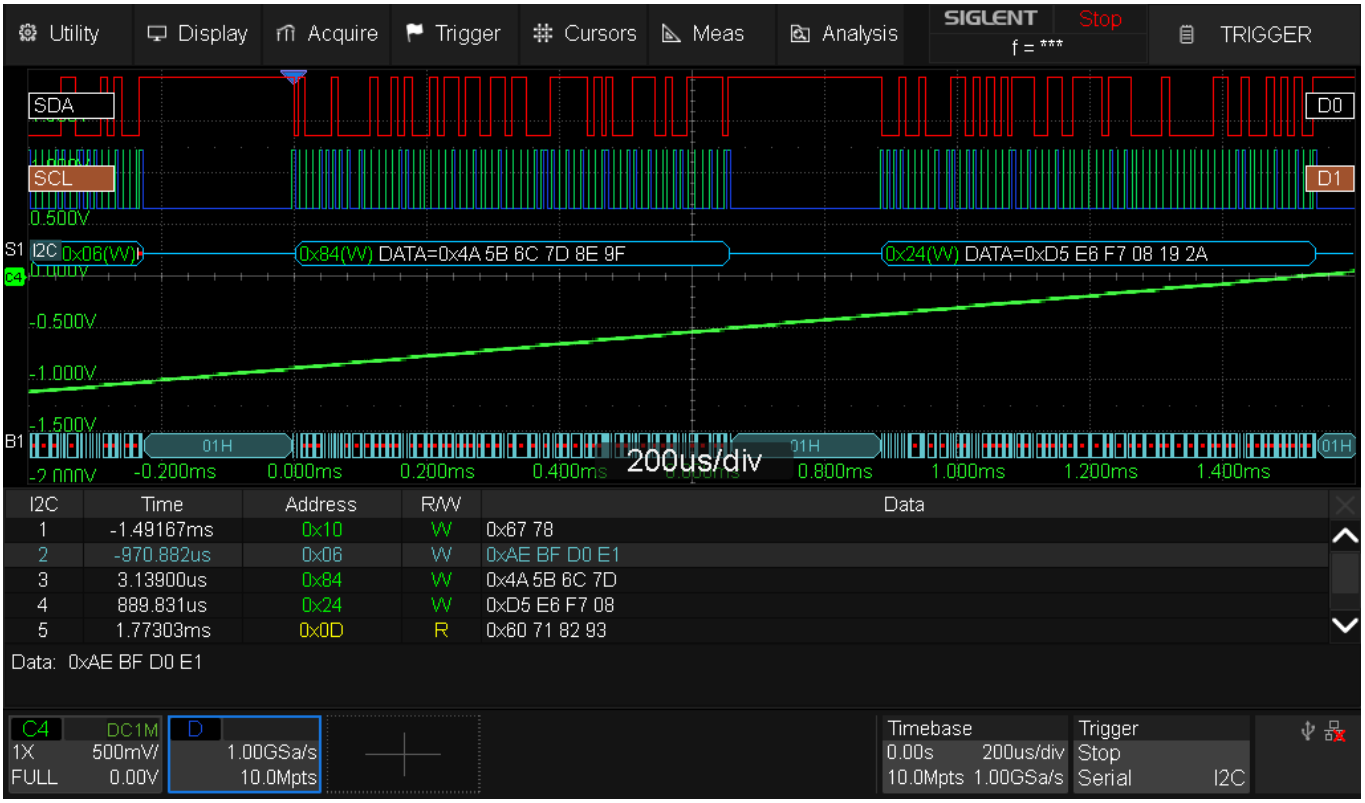

Fig. 166 SDS824X_HD_Digital_1Ch_I2C_Stop_Zoom_200us

There's finally the question: what if we want some advanced measurements (with statistics and Histicons) on top of that all?

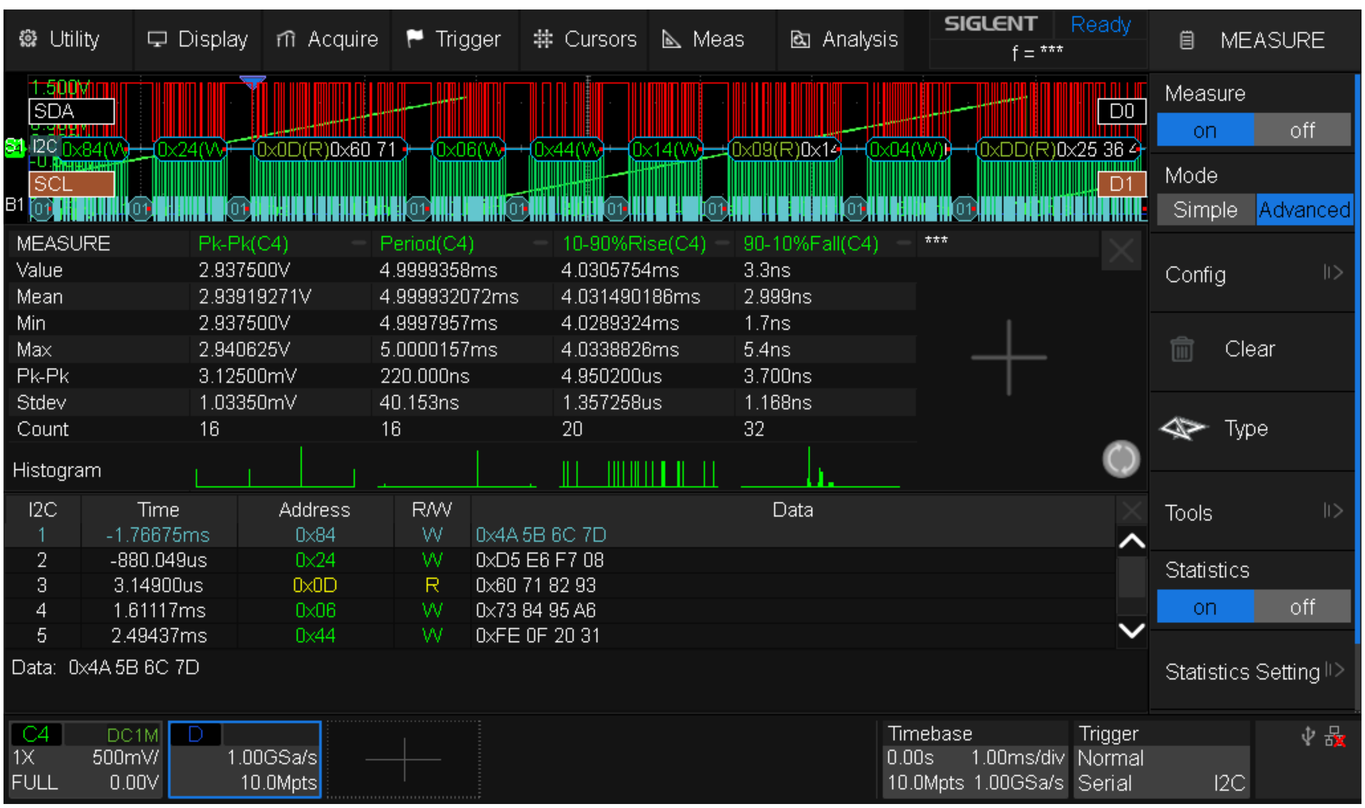

Fig. 167 SDS824X_HD_Digital_1Ch_FullHouse

Well, this looks a bit cramped, yet still not so bad if only we'd disable that useless parallel bus decoder line. Maybe we could make do with simple measurements, then we'd get a bigger signal display again.

Serial Decoders

This is just a quick test to demonstrate some serial decoders. For convenience's sake (simple connections to the test board without having to bother with probes), I've used the digital channels of the SLA1016.

The I2C decoder has already been used in chapter "Digital Channels", so I'll not show it again here.

First, there's some SPI data stream with simple edge trigger on /CS:

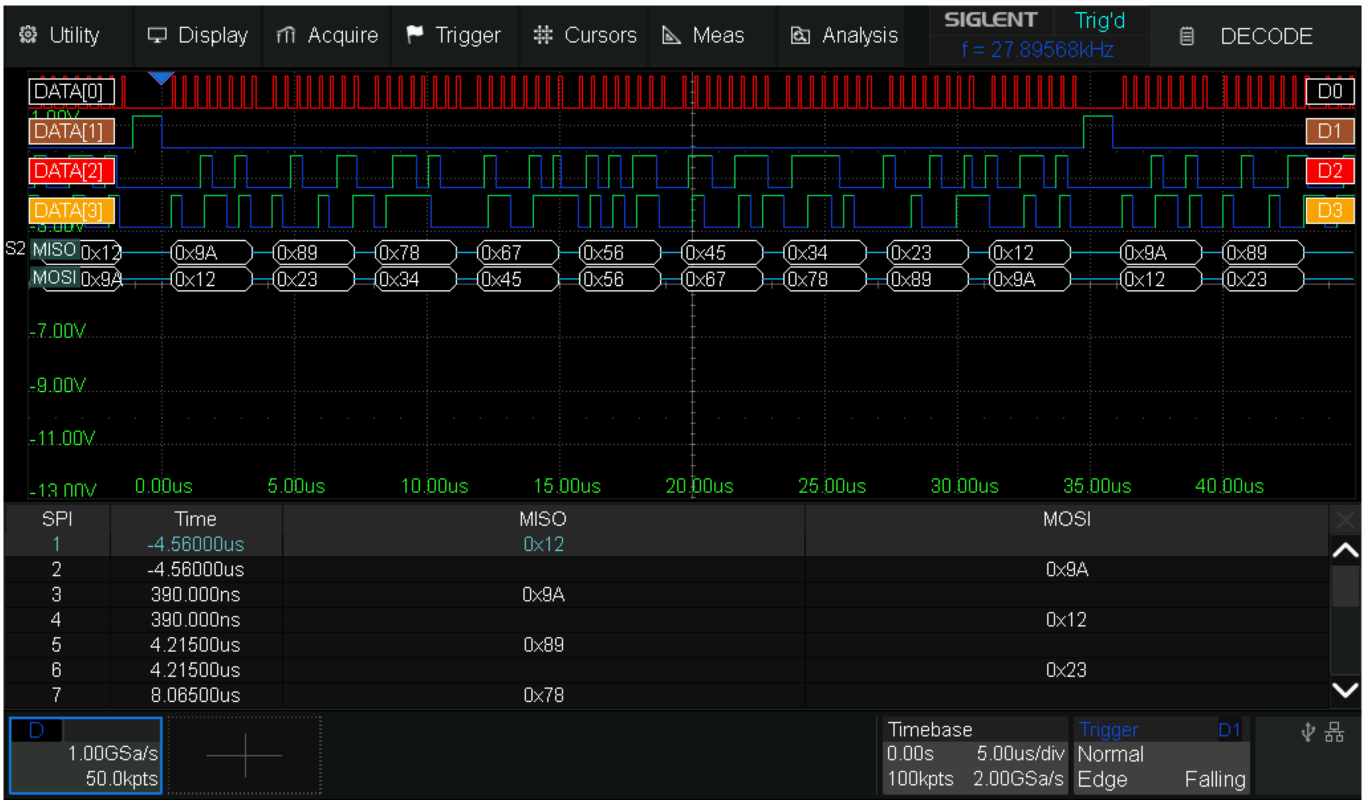

Fig. 168 SDS824X HD_Digital_SPI

This is some UART data stream in 8N1 format with serial UART trigger on the start condition of a byte:

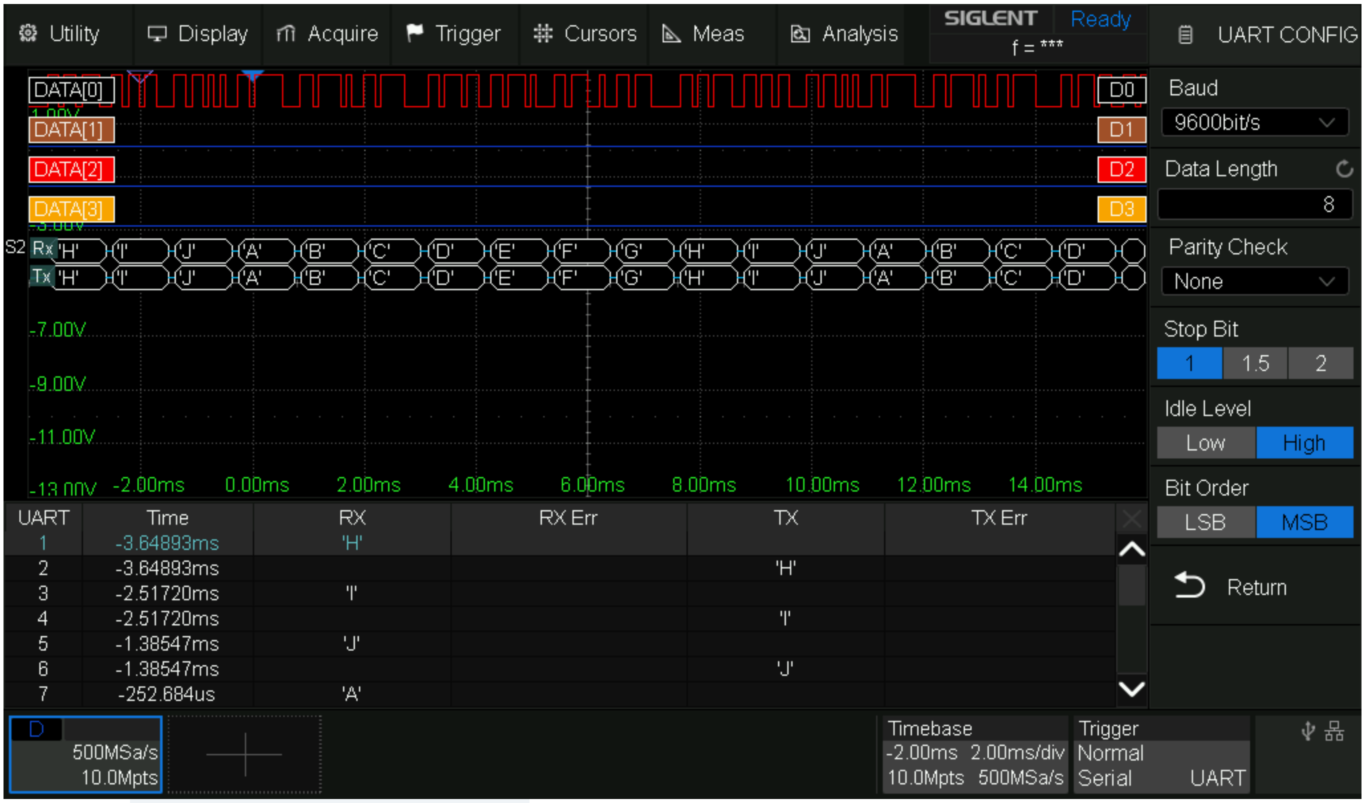

Fig. 169 SDS824X HD_Digital_UART

Finally, a CAN telegram with serial CAN trigger on the start condition.

Fig. 170 SDS824X HD_Digital_CAN

HAM Test

Here come some tests that might be of interest for HAM operators.

First, I should ensure that expectations don’t get unrealistic. A general-purpose oscilloscope like the SDS800X HD is neither a spectrum analyzer nor a test receiver, even when its noise figure drops below 10 dB at frequencies above 300 kHz. Yet it just works on the full input signal bandwidth, there is no pre-selector, no mixer to shift small portions of the upper spectrum down to a lower intermediate frequency and no filter that could isolate the IF signal.

In the time domain, 400 μV_{RMS} is pretty much the limit for reliable triggering, indicated by a correct trigger frequency counter display; for even lower levels, triggering gets increasingly sporadic and at 100 μV_{RMS}, the trigger frequency counter might drop below 1 MHz. Only if we have a strong copy of the signal of interest, such as the generator signal when measuring the stop band of a filter, we could trigger on the input signal and use the Average math function to make even very weak output signals visible and measurable, even those way below the noise floor, as has been demonstrated already the “Zoom Expectations” chapter.

For all the following tests, a Siglent SDG7102A AWG with OCXO option has been directly connected with coaxial Hyperflex 5 cables, a step-attenuator Wavetek 5080.1 and an inline terminator “hp10100C” at the scope input.

CW Test

For the 1 mV_{RMS} demo I had to bring quite a lot of information onto a single screen:

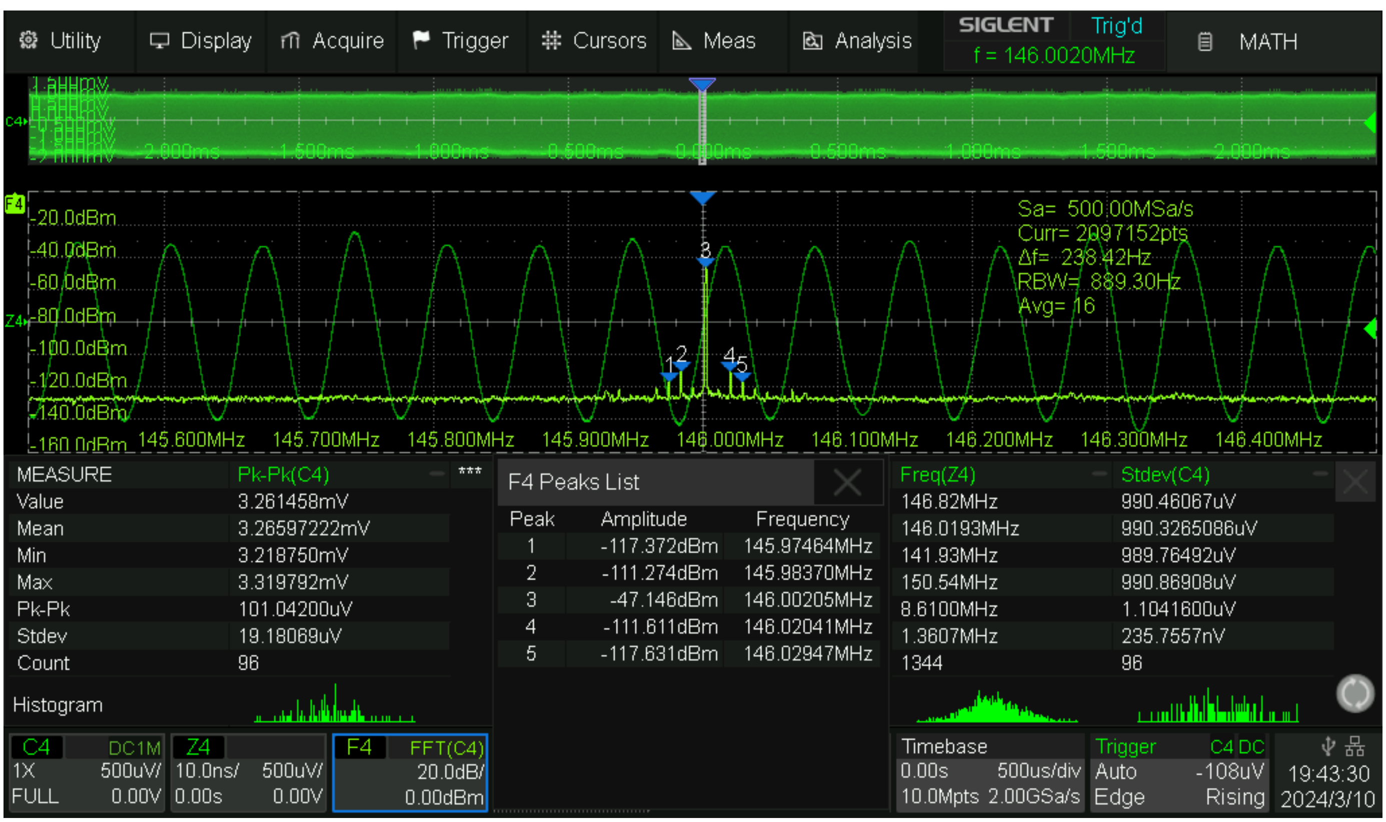

Fig. 171 SDS824X HD_SA_CW_146MHz_1mV

The main window at the top is showing the 146 MHz signal at a time base of 500 µs/div. We need that slower time base to get sufficient data for a 2 Mpts FFT. Of course we can't see any details there, hence there is the zoom window below, where the signal is displayed at a time base of 10 ns/div. We can see that the vertical position is not constant across the screen width, which hints on the noise that gets very noticeable at such low signal levels.

The zoom window also contains the FFT which reveals some weak modulation due to noise and we can see the corresponding sidebands indicated by markers 1, 2, 4 and 5, whereas marker 3 corresponds to the main signal.

Below we have two windows nested: the advanced measurements together with the Peaks List, for which I've freed up some space by removing measurements 2 and 3. We can see that the amplitude measurements in the time domain still work reasonably well, at least Stdev (= AC-RMS), whereas the peak-to-peak amplitude cannot be accurate – once again because of the noise.

The signal frequency measurement is not accurate as well and all automatic measurements have a high variance, as can be seen in the measurement statistics and histograms, yet the 7-digit trigger frequency counter is still rock stable and pretty accurate. Of course, it is 2 kHz off, which is equivalent to 13.7 ppm, but that's because the time base of the SDS800X HD has 25 ppm tolerance, which is pretty much standard, while Siglent's higher end DSOs starting at the SDS2000X plus provide class leading 1 ppm time base accuracy.

The peak table shows the various signal amplitudes, but we are mainly interested in Peak #3, which is indicated as -47.146 dBm at 146.00205 MHz. Surprisingly, the frequency is exactly the same as the trigger frequency counter, hence absolutely correct relative to the SDS800X HD time base, also the amplitude is pretty much spot on (should ideally be -47.0 dBm).

At this point I should mention that the Peak table is automatic and we only set the search parameters, i.e. threshold and excursion. This is in contrast to the Markers, which can be preset on peaks or harmonics, yet each marker can be set individually to any desired frequency by the user. For the measurements here, Peak tables have been used exclusively.

There is no point to further look at the time domain for weaker signals, because they will get heavily distorted and eventually buried in the noise anyway. Yet I don't change the arrangement and only move the zoom trace out of view.

Next, we try 100 μV_{RMS}:

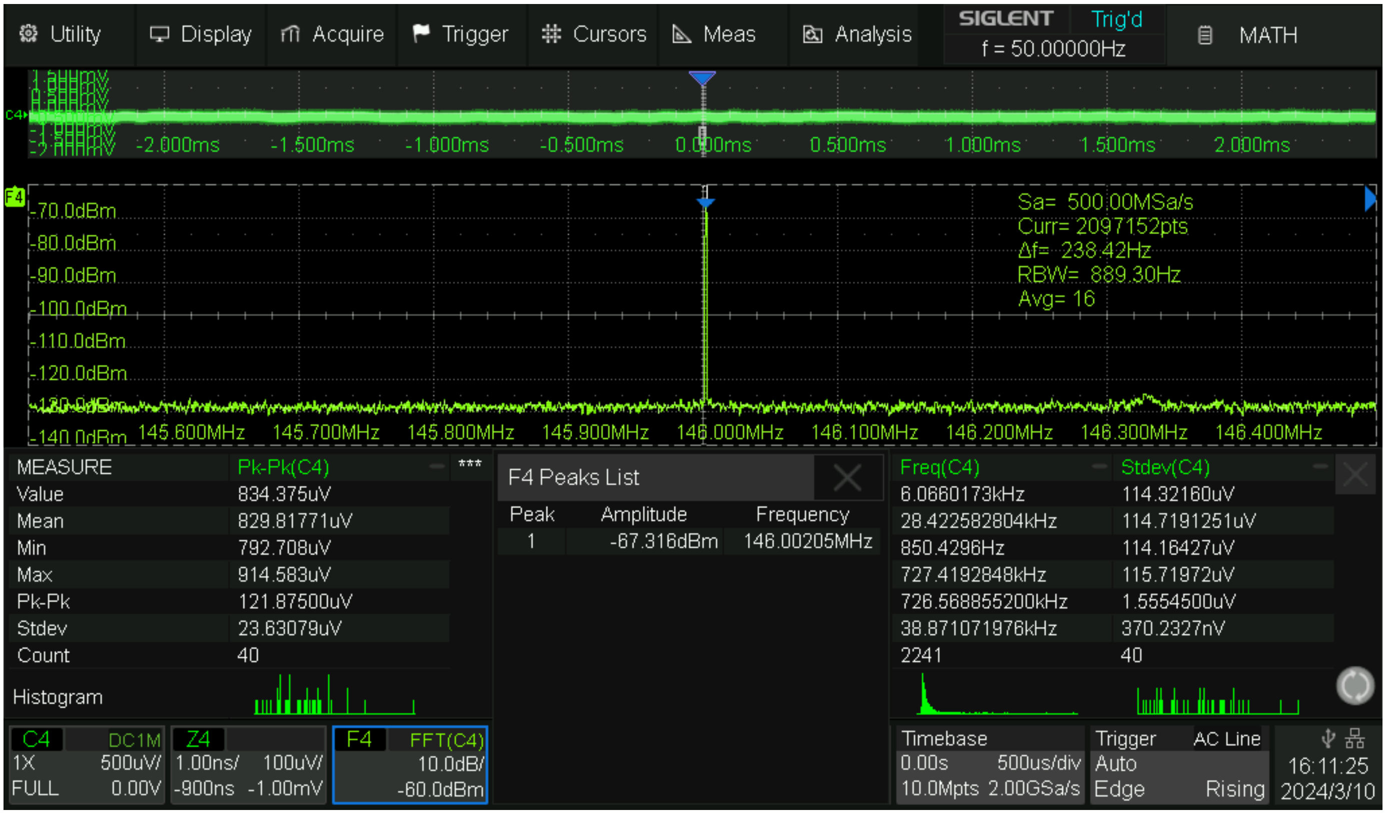

Fig. 172 SDS824X HD_SA_CW_146MHz_100uV

The main window at the top shows predominantly noise for the 100 µV_{RMS} signal at 245 MHz bandwidth. The zoom window with the FFT displays a clean spectral line, hence we only get a single peak marker.

The advanced measurements below just try to measure the noise, even though the Stdev measurement still isn't too far off – just a coincidence?

Since such a weak signal cannot be properly triggered anymore anyway, I've used AC-line trigger instead and the trigger frequency counter just shows a constant 50 Hz now. FFT doesn't need a triggered signal.

The peak table shows the signal amplitude as -67.316 dBm at 146.00205 MHz. The amplitude is still pretty accurate (should ideally be -67.0 dBm).

Now let's try 10 µV_{RMS}:

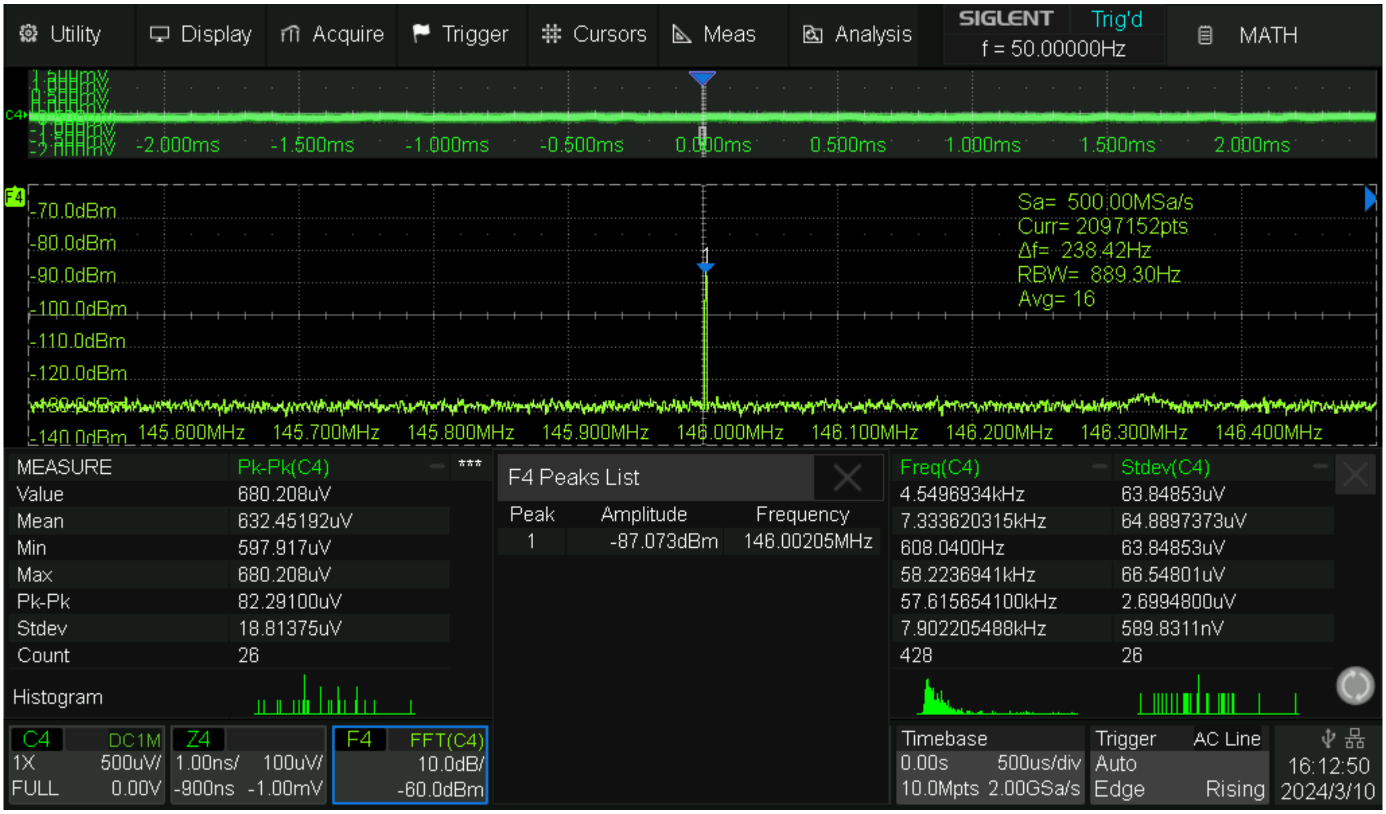

Fig. 173 SDS824X HD_SA_CW_146MHz_10uV

Once again, we get a clean spectral line with a single peak marker.

The advanced measurements below are totally meaningless, but the peak table shows the signal amplitude extremely accurate as -87.073 dBm at 146.00205 MHz.

Another attempt with 1 μV_{RMS}:

Fig. 174 SDS824X HD_SA_CW_146MHz_1uV

One more time, we get a clean spectral line with a single peak marker.

The advanced measurements below are totally meaningless, but the peak table shows the signal amplitude pretty accurately as -107.291 dBm at 146.00205 MHz.

A final attempt with 100 nV_{RMS}:

Fig. 175 SDS824X HD_SA_CW_146MHz_100nV

We even get a tiny spectral line, but have to tweak the search parameters to get the peak marker. Its amplitude is now way off: -123.737 dBm at 146.00182 MHz (should be -127.0 dBm). The signal is just too close to the noise floor now.

This test is quite revealing. Even a 100 nV_{RMS} signal could be detected, and even though the signal amplitude was off by 3.26 dB, the frequency could still be measured with just ~200 Hz error (relative to the DSO's time base that is)!

FM Test

For the modulation tests, I did not feel like trying many different levels again, so 1 mV and 10 µV RMS shall be sufficient for FM.

First, I need to repeat what I've already stated initially: An oscilloscope is a baseband instrument, not a spectrum analyzer. There is only so much frequency resolution we can get from a 2 Mpts FFT. As a consequence, the expectations shouldn't be very high when looking at one kilohertz frequency spacing in signals from the 2 meter band.

Modulation: 1 kHz modulation frequency and 5 kHz frequency deviation.

1 mV_{RMS} at 146 MHz first:

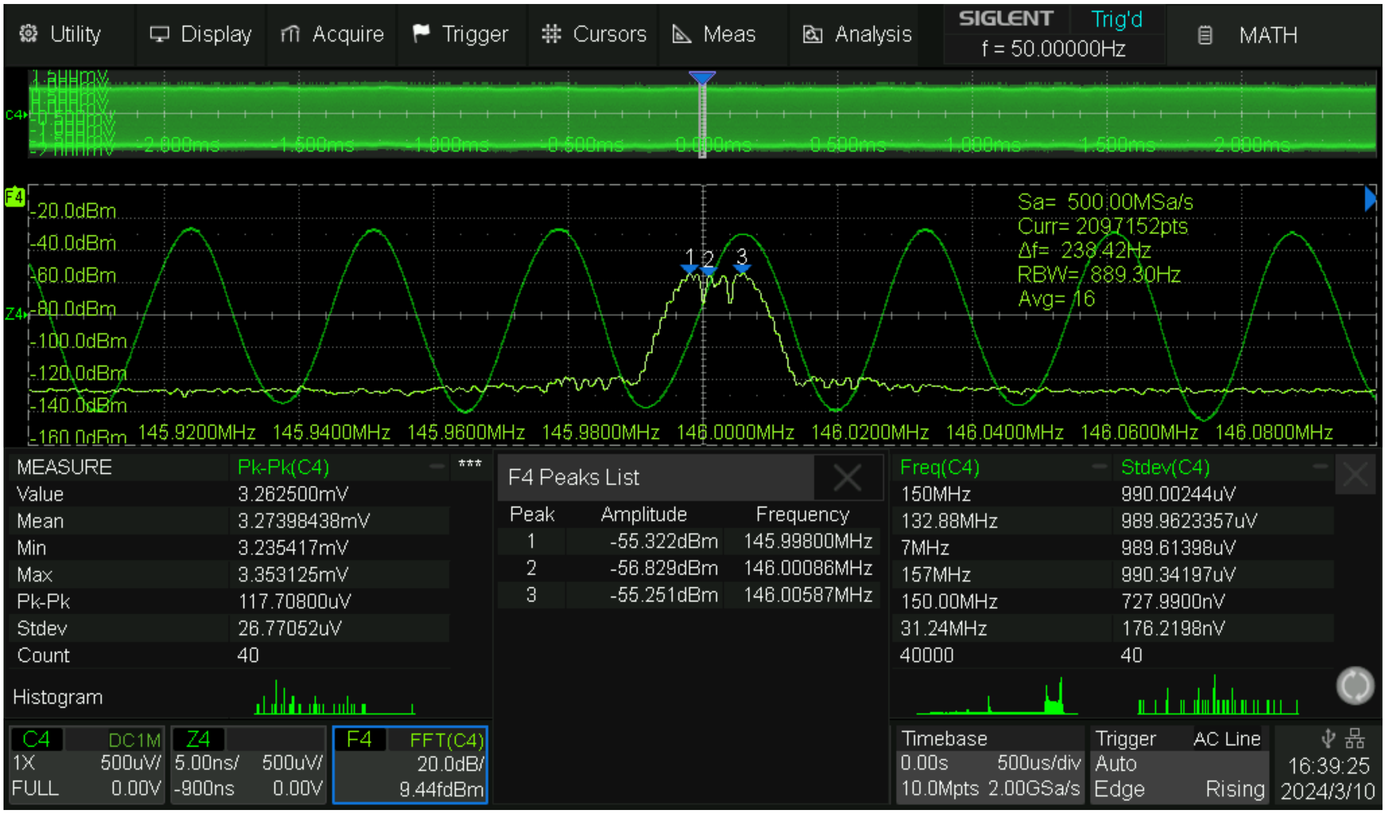

Fig. 176 SDS824X HD_SA_FM_146MHz_1mV

Well, that's not very convincing. I was forced to use the Blackman window, which is still usable for spectrum analysis with a max. amplitude error of 1 dB. Its resolution bandwidth is not that much better than Flattop hence the picture isn't very useful and we only get a vague idea of the FM spectrum.

It should be clear that a general-purpose oscilloscope isn't a test receiver. If we look at bandwidth limited signals, like filtered IF, where we need not worry about aliasing from unrelated signals in the neighborhood, we could just use the "frequency conversion by undersampling" method.

Any ADC acts as a mixer, thus producing a spectrum of $\pm n * fi \pm m * fs$, where fi is the input frequency and fs is the sample clock, while n and m are just integers running from 0 to (theoretically) infinity. During normal operation, we don't want to see any mixer products, which is perfectly possible as long as the input signal and all its harmonics don't exceed $fs/2$ and the output of the ADC has a brick-wall filter (then in the digital domain of course, aka Sinc filter or sin(x)/x reconstruction) that removes everything above $fs/2$.

Yet in some circumstances, we can make use of a certain high-order mixer product, just as in this example, where the effective FFT sample rate is only 5 MSa/s, which is quite obviously way too low for a 146 MHz input signal.

According to the formula given above, we are aiming at the mixer product for $1 * fi - 29 * fs$, which is

1 * 146 MHz – 29 x 5 MHz = 146 MHz – 145 MHz = 1 MHz;

Now if we set the center frequency to 1 MHz, we can look at the 146 MHz carrier with a resolution bandwidth of only 35.57 Hz, even with just 512 kpts FFT, hence can also clearly see the sidebands and their 1 kHz spacing. 16x averaging has been used in order to get a clean and stable display:

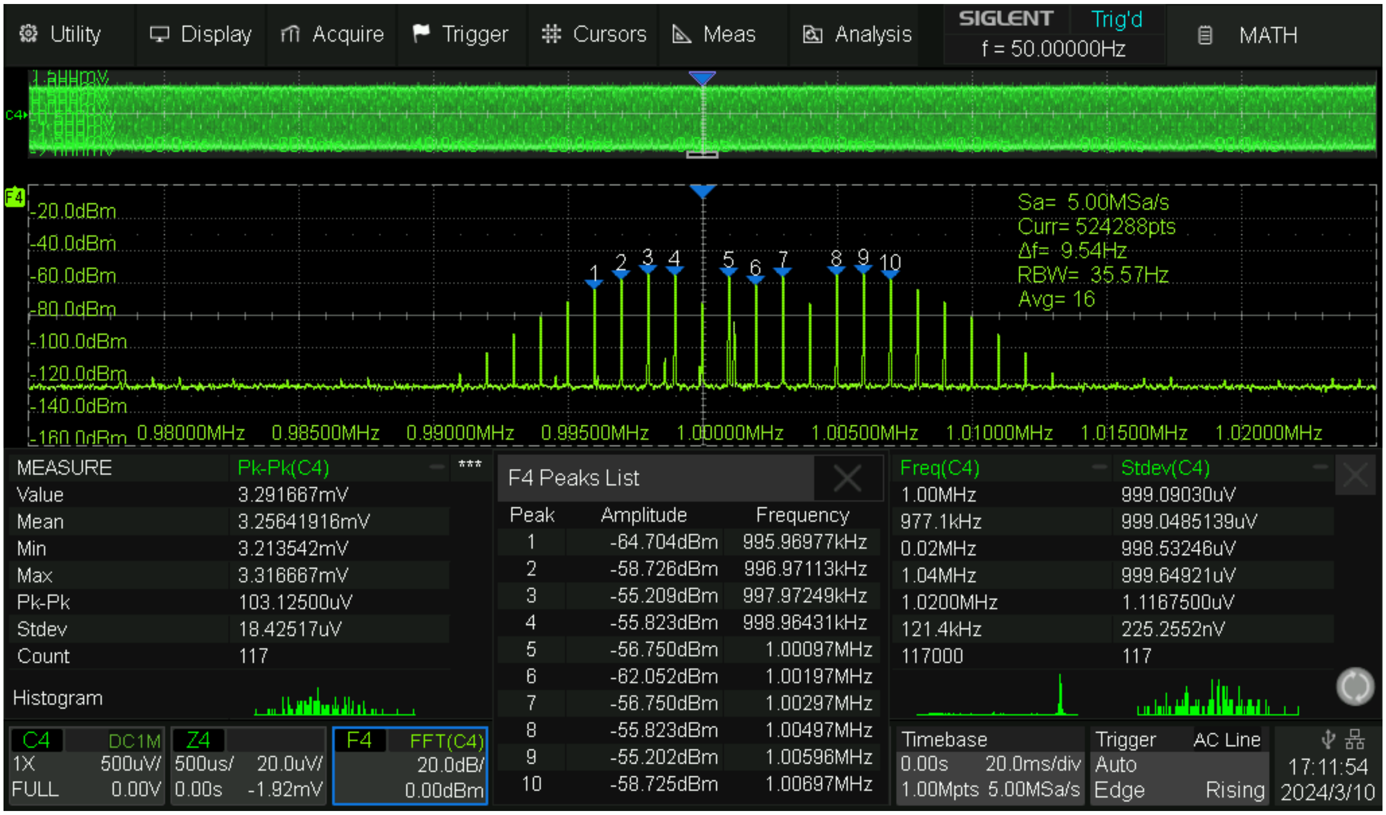

Fig. 177 SDS824X HD_SA_FM_146MHz_1mV_5MSa

Be aware that the true center frequency is Peak #6 at 1.00197 MHz because of the SDS824X HD time base tolerance of 25 ppm.

The following table compares the measured sidebands with the expected ones in dBm:

Order	Measured	Expected
0	-62.052	-62.083
1	-56.750	-56.763
2	-73.700	-73.713
3	-55.823	-55.833
4	-55.200	-55.223

Now for the 10 μV_{RMS} test:

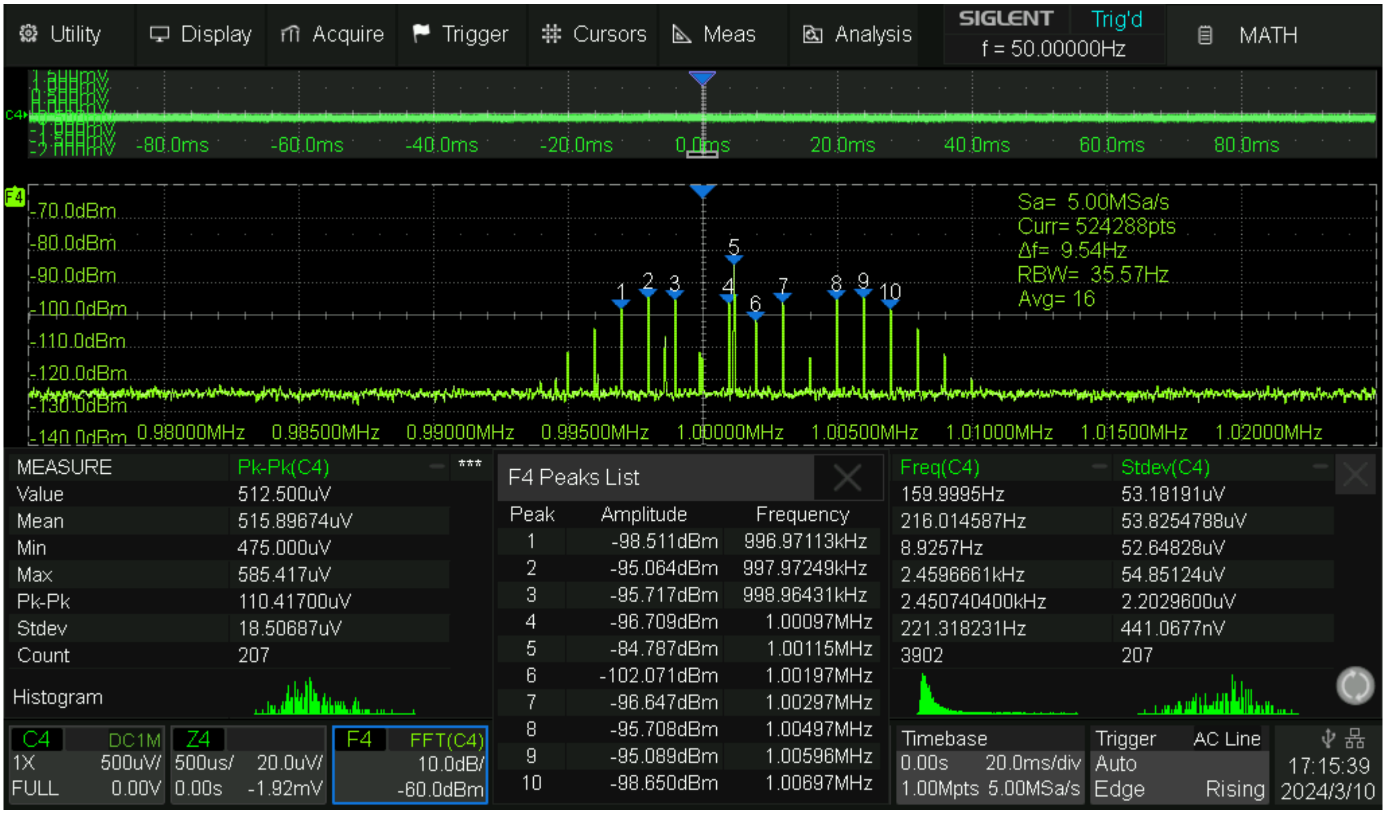

Fig. 178 SDS824X HD_SA_FM_146MHz_10uV_5MSa

There is a spurious signal at marker 5 exceeding the signal spectrum – we just have to ignore it.

The following table compares the measured sidebands with the expected ones in dBm:

Order	Measured	Expected
0	-102.071	-102.083
1	-96.700	-96.763
2	-113.700	-113.713
3	-95.700	-95.833
4	-95.100	-95.223
5	-98.500	-98.733

All in all, the down-mixing via the ADC can save our day as long as we have an isolated signal and the signal levels don't get too low.

There are several caveats though:

- We need to make sure that *n* is always positive, otherwise we'd get the result in reverse frequency position, i.e. the upper sideband appears below the carrier and vice versa.
- Mixing with the 29th harmonic of the sample clock introduces also 29 times more phase noise and jitter and this might get visible the FFT plot.
- Amplitude accuracy might suffer, as a harmonic mixing process is not guaranteed to be as efficient as the fundamental one, hence we might see some attenuation.

AM Test

Modulation: 1 kHz modulation frequency and 80% modulation depth.

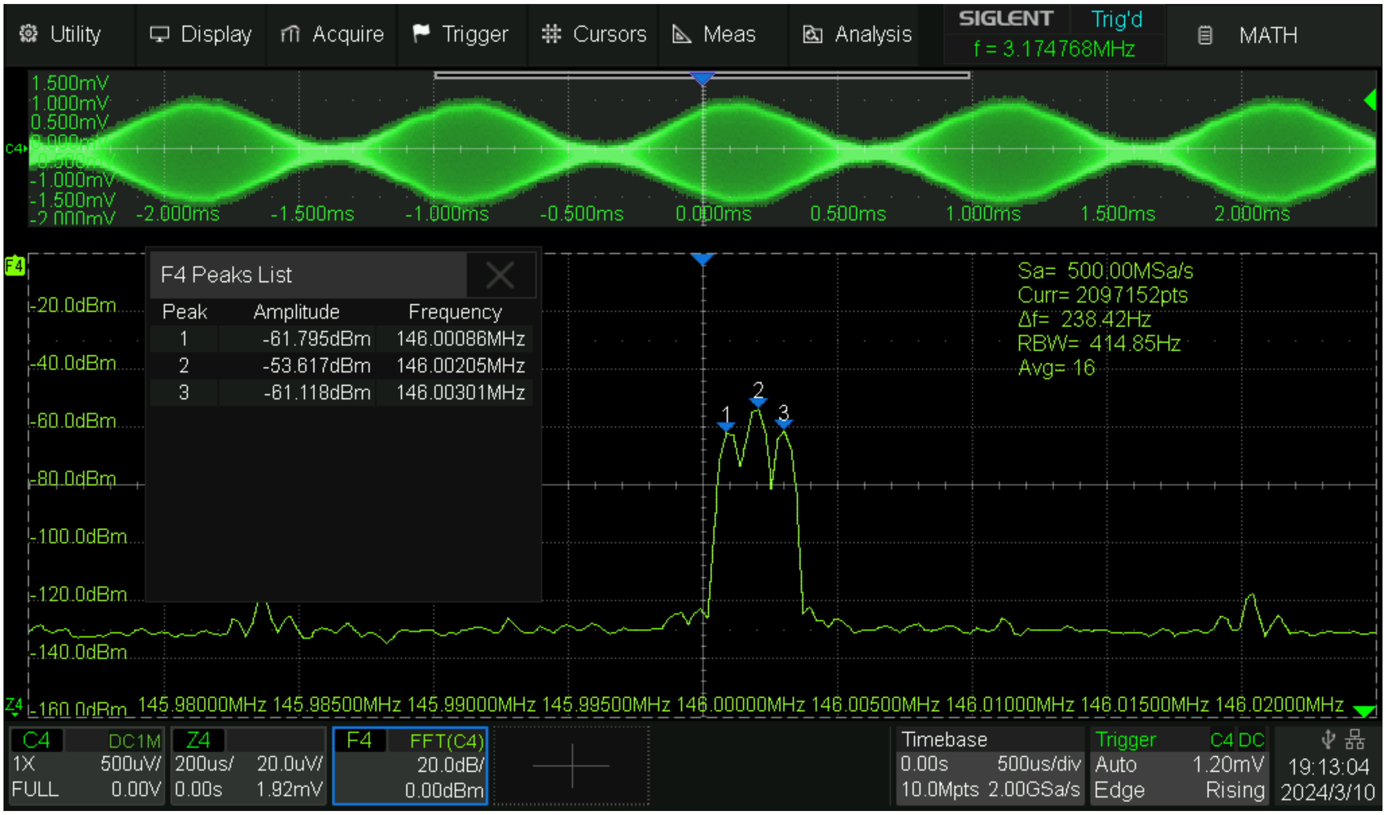

Fig. 179 SDS824X HD_SA_AM_146MHz_1mV

Well, that looks familiar. Once again, the picture isn't great and we only get a vague idea of the AM spectrum. The sidebands should be -8 dBc = -61.6 dBm. The deviations come from the insufficient resolution bandwidth and the amplitude error of the Blackman window.

We are looking at the mixer product for 1 * *fi* - 29 * *fs* again, which is

1 * 146 MHz – 29 x 5 MHz = 146 MHz – 145 MHz = 1 MHz;

Now if we set the center frequency to 1 MHz, we get the carrier at 146 MHz and a resolution bandwidth of only 35.57 Hz, hence can also clearly see the sidebands and their 1 kHz spacing. 16x averaging has been used in order to get a clean and stable display:

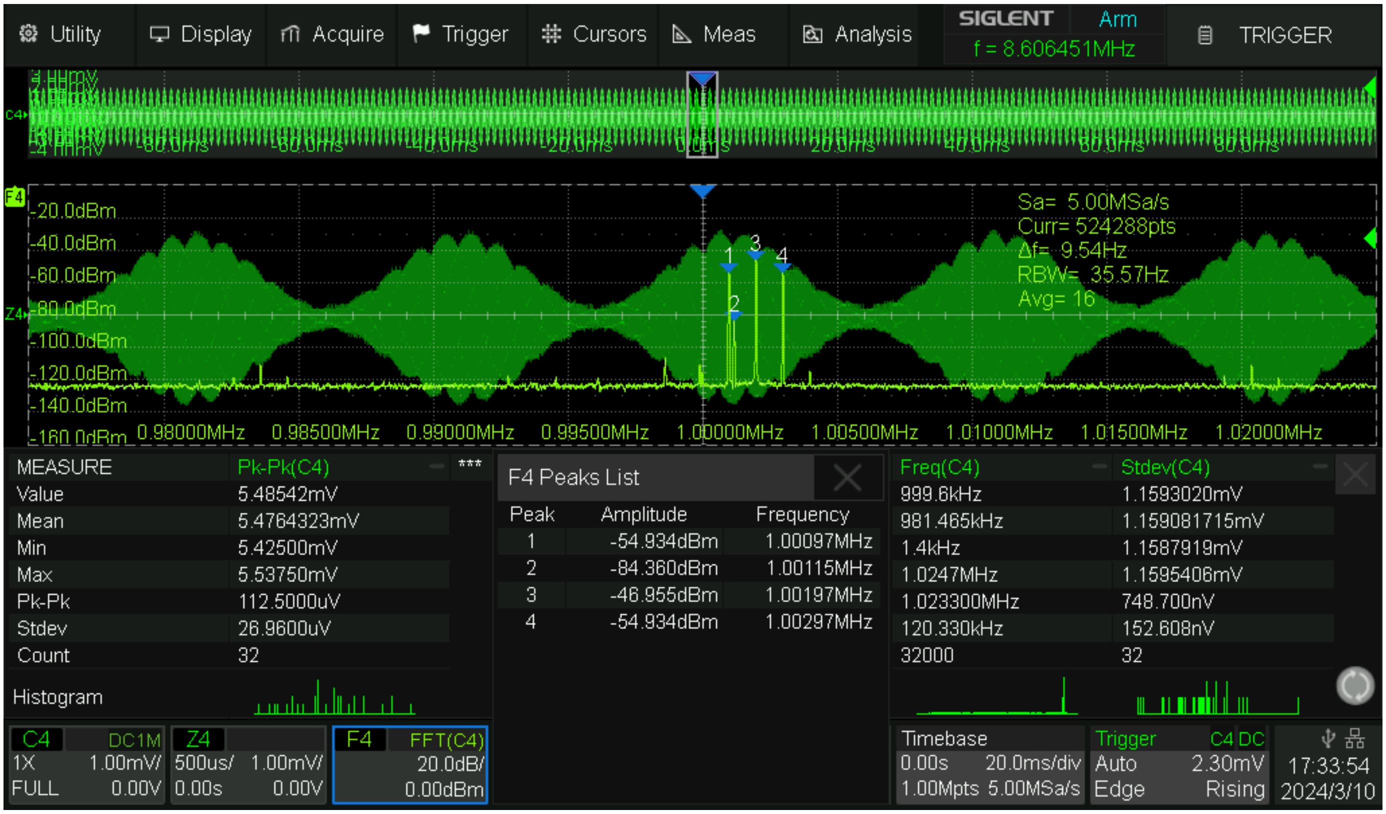

Fig. 180 SDS824X HD_SA_FM_146MHz_1mV_5MSa

Be aware that the true center frequency is Peak #3 at 1.00197 MHz because of the SDS824X HD time base tolerance.

The sidebands should be -8 dBc = -55 dBm. The accuracy is very high despite the harmonic mixing process.

Filter Demo

There was the suggestion for cleaning up high frequency narrowband signals by means of a bandpass filter. In this case we are talking about the 146 MHz signal from the HAM Test demo.

The narrowest bandwidth achievable at 146 MHz is 40 MHz (126-166 MHz). This appears very wide, but is pretty effective nevertheless. Just look at the noise floor:

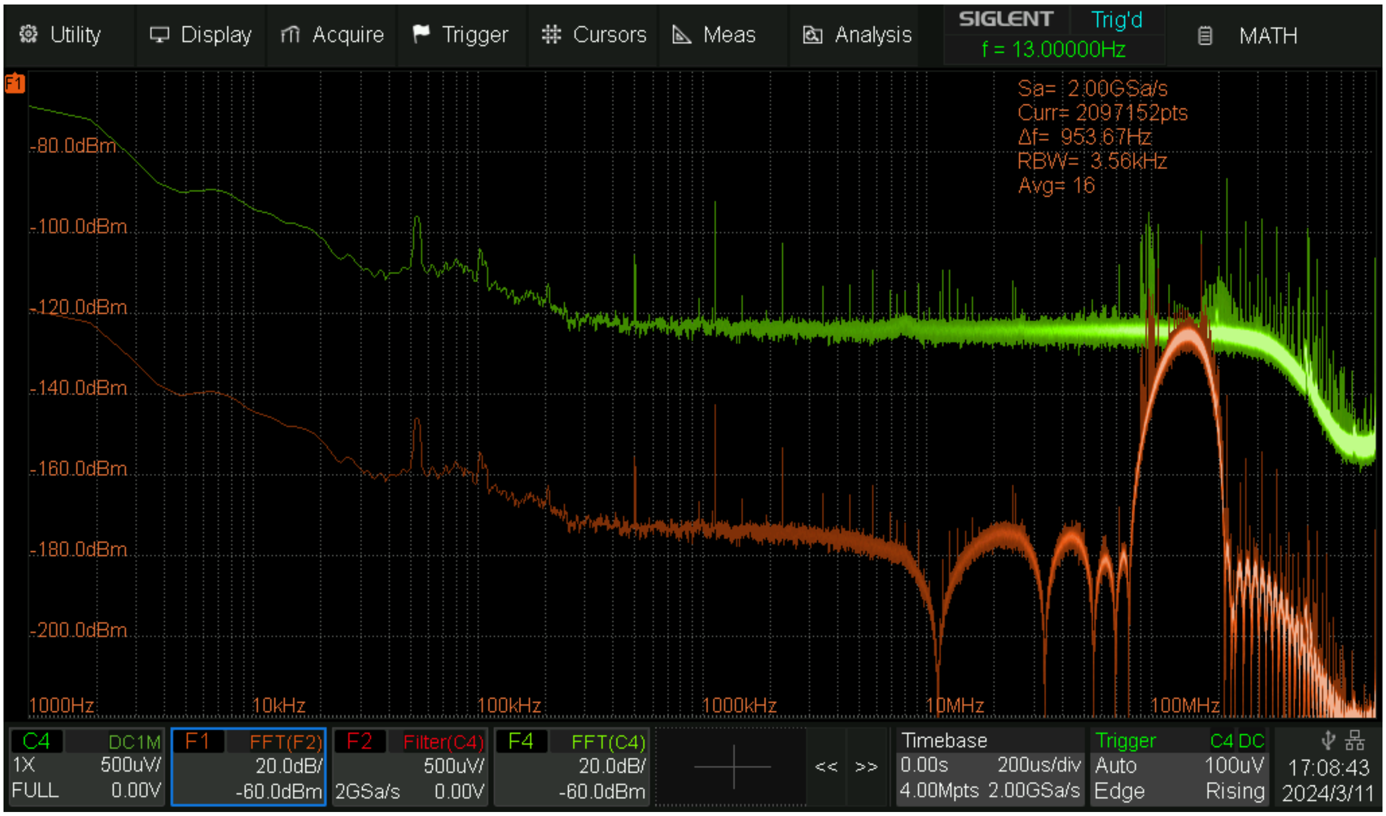

Fig. 181 SDS824X HD_Math_FFT_BP_126-166MHz_Noise

The screenshot above shows two math channels with FFT plots; F4 is the original signal (which is just the frontend noise for now), whereas F1 shows the result of the filter operation performed in math channel F2. It should be immediately obvious that getting rid of the strong 1/f-noise below 300 kHz alone would help a lot. With the BP-filter, only at 1 kHz the 1/f-noise approaches the level of the original HF-noise (~-120 dBm).

We cannot expect a major improvement for FFT measurements, because the resolution bandwidth there is much narrower than our bandpass-filter anyway. Here is the FFT with the 1 mV_{RMS} signal at 146 MHz applied:

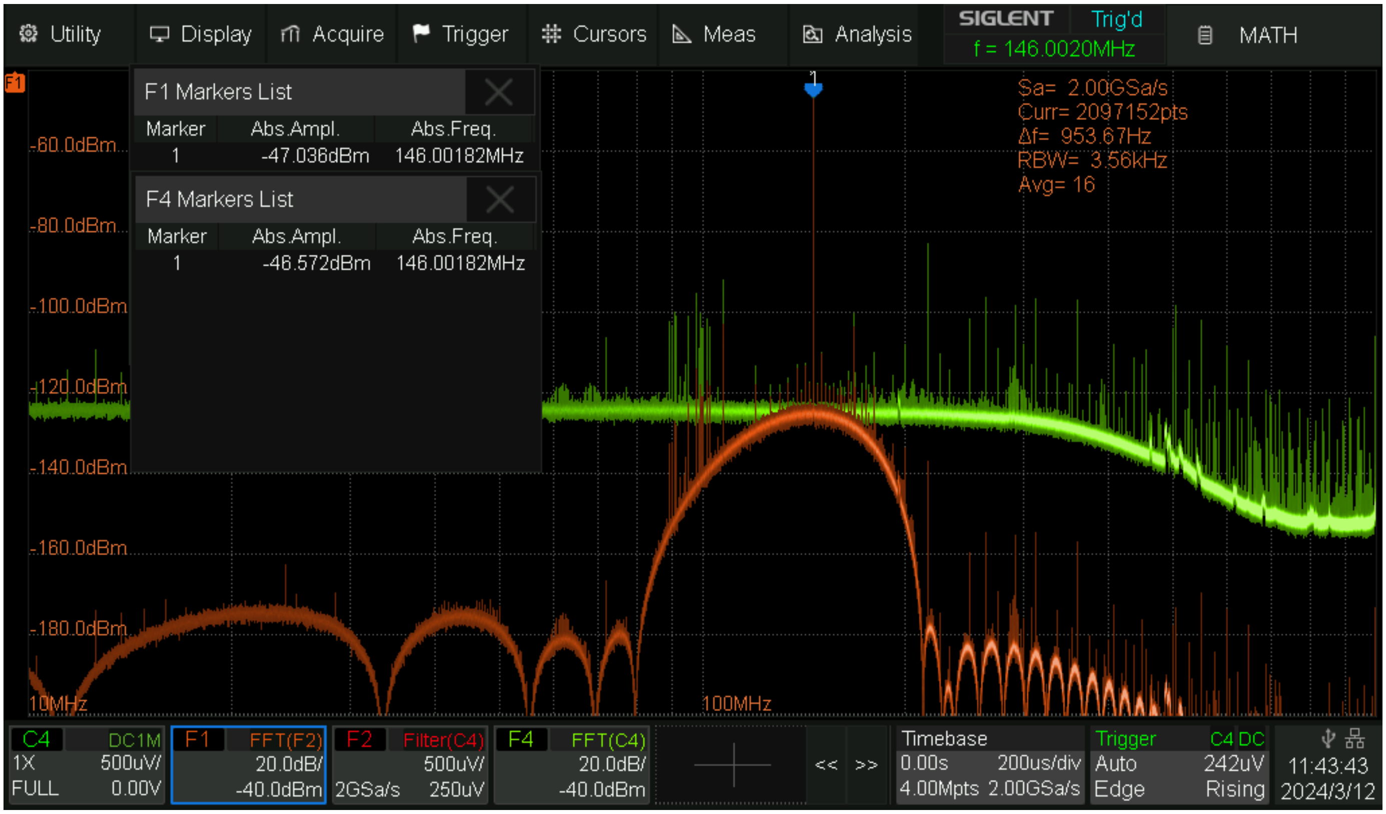

Fig. 182 SDS824X HD_Math_FFT_BP_126-166MHz_1mV

Yes, there actually is a difference. The original signal is measured ~0.43 dB too high because of the noise, resulting in -46.57 dBm, see F4 Markers List. On the other hand, the filtered version of that signal measures -47 dBm almost spot-on, as can be seen from F1 Markers List. So yes, even the FFT measurements can benefit from a cleanly filtered signal.

Now let's have a look at the time-domain, where the improvement because of the filter should be even more obvious:

Fig. 183 SDS824X HD_Math_BP_126-166MHz_1mV

Advanced automatic measurements have been set up for the amplitudes of both the original signal and its filtered version. The visual difference is quite striking already, and the measurements confirm the improvement: just like with the FFT, automatic measurements are more accurate with filter than without.

We already know that reliable triggering is only possible down to about 400 μV_{RMS}, yet it still works with 100 μV_{RMS} signals, if we can accept a high rate of missed triggers (just look at the trigger frequency counter):

Fig. 184 SDS824X HD_Math_BP_126-166MHz_100uV

Once again, the visual difference is quite striking, and even though the automatic measurements for both signal variants aren't entirely accurate, there is still a major improvement with the filtered version.

User Comments

Fun with Square Waves

Comment by BRZ.tech

I have another question that I consider relevant to the HAM universe, which is the Frequency Response of the SDS800X HD for Square Wave Excitation.

…

And the analysis of Digital Signals in the time regime and frequency regime is very welcome for HAM users. And I'm sure the SDS800X HD will literally be able to supply the most important measurements.

In this topic, I looked in the index, in the first post, and I didn't find the Frequency Response for Square Wave Excitation.

The frequency as well as pulse response of the SDS800X HD have been presented in some detail in the sections “Bandwidth” and “Pulse Response”). Why do you think you need a special frequency response for square waves?

You’re talking about Fourier and point to a video showing how to derive the well-known Fourier coefficients for a square wave. Maybe these all too theoretical approaches still don’t convey the fundamental insight that the fidelity of a square wave reproduction on the oscilloscope screen depends on the rise time of the square wave and the bandwidth of the scope. The scope bandwidth is known, whereas our task now is to find the rise time a 200 MHz DSO can handle smoothly.

Okay, let’s recap some fundamentals.

Even with moderate 3.5 ns rise time, the pulse spectrum extends up to >800 MHz – look at the spectrum in the screenshot below (for this I pulled out a heavier gun with 2 GHz bandwidth):

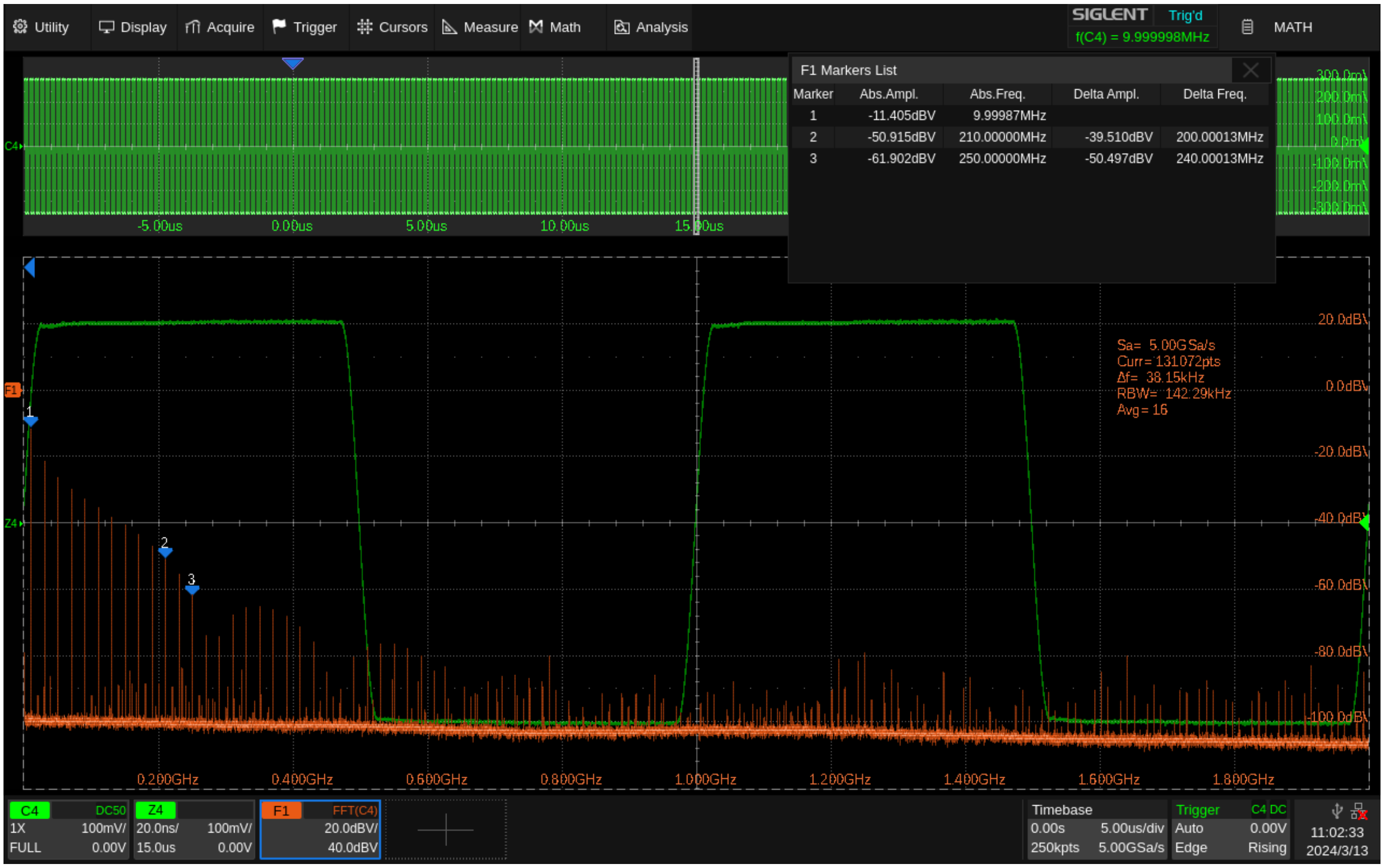

Fig. 185 Ref-Spec_Square_3.5ns_10MHz

Of course, a fast scope like this reproduces a comparatively slow rise-time like 3.5 ns flawlessly. We can see that with 210 MHz bandwidth, we could expect a decent reproduction of this signal with no more than ~1% aberrations, because all harmonics down to -40 dBc would be included. With 250 MHz bandwidth (which the SDS824X HD nearly has with up to 2 channels in use), the expected aberrations would be even lower at about 0.3%, as all the harmonics down to -50 dBc would be included.

Comment by BRZ.tech

I suggest you do a test to measure a signal, which we can use in HAM, a SQUARE WAVE signal f=50.000MHz, without modulation, Amplitude = 0 to 1V_{PP}, at 50 Ohm, to start, and increase the frequency, to have the minimum number of tracks that you think are necessary in the FFT of the SDS800X HD, for the cutoff frequency that you consider sufficient for the BW.

Well, let's see what a 50 MHz square wave would look like if we stick to a rise-time of 3.5 ns, which would be adequate for the SDS824X HD:

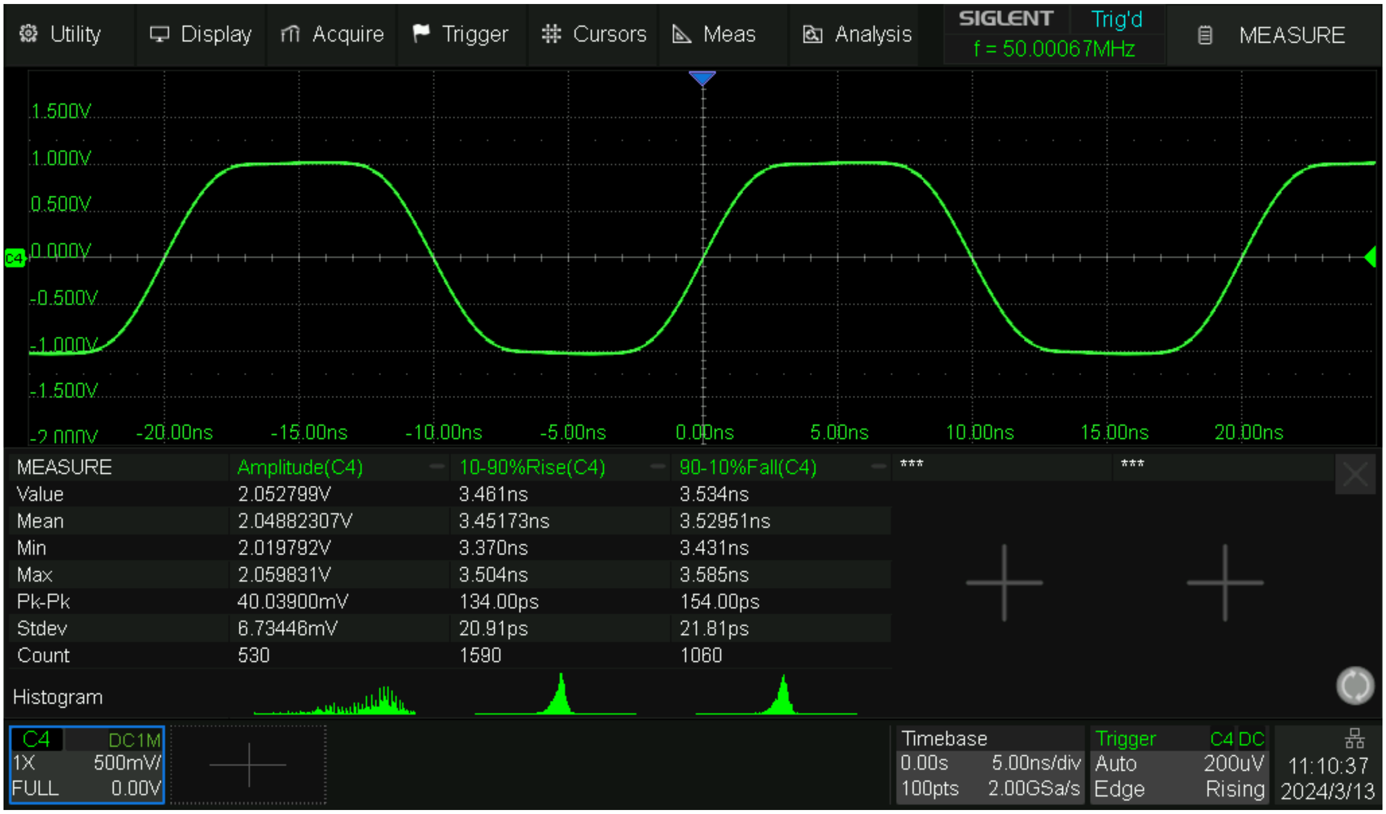

Fig. 186 SDS824X HD_Square_3.5ns_50MHz

Yes, this is a perfectly flawless rendering of the square wave with 3.5 ns rise time, as also quite accurately measured by the SDS824X HD. A Look at that picture enables us to predict how an even faster square wave would look; here's an example for 80 MHz:

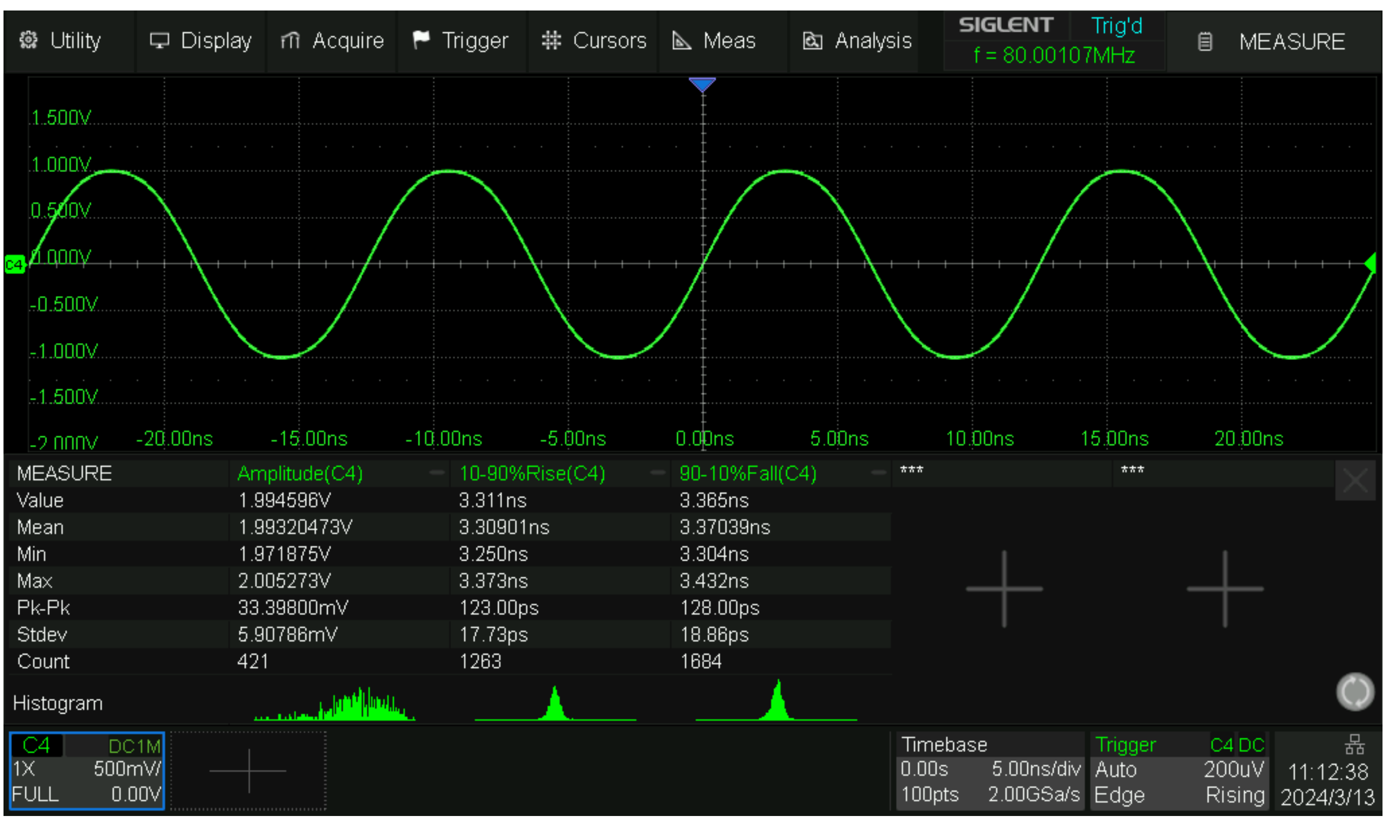

Fig. 187 SDS824X HD_Square_3.5ns_80MHz

Right, there's almost a pure sine left. With a 3.5 ns rise time, we cannot expect anything better.

Of course, we can use a faster rise time, like 1 ns, which clearly is a bit of a challenge for a 200 MHz DSO:

Fig. 188 SDS824X HD_Square_1ns_80MHz

Overall, the result doesn't look bad – it just can't properly characterize the original signal as the rise time measurement is totally off and approaches that of the DSO itself. The signal shape is just what to expect from a bandwidth limited square wave signal. The amplitude measurement is still pretty accurate though.

I hope it is now clear that it's not the frequency of a square wave, but its rise time, which makes all the difference.

Comment by BRZ.tech

At the end of the test, the "Fourier Series Coefficients" collected from the "FFT Table" of the SDS800X should be very close to those presented by "Professor Michel van Biezen".

There is the question that how the "Fourier Series Coefficients" in their Amplitude values should be presented in the f(t) equation: V_{PP}, V_{RMS}, V_P, or other??

I did not watch that video to the end, sorry. The Fourier series for standard waveforms are well-known, so we need no educational video for that.

Let's start with the reference again. What does the 80 MHz square wave with 1 ns rise time look on a DSO that is actually fast enough to handle it?

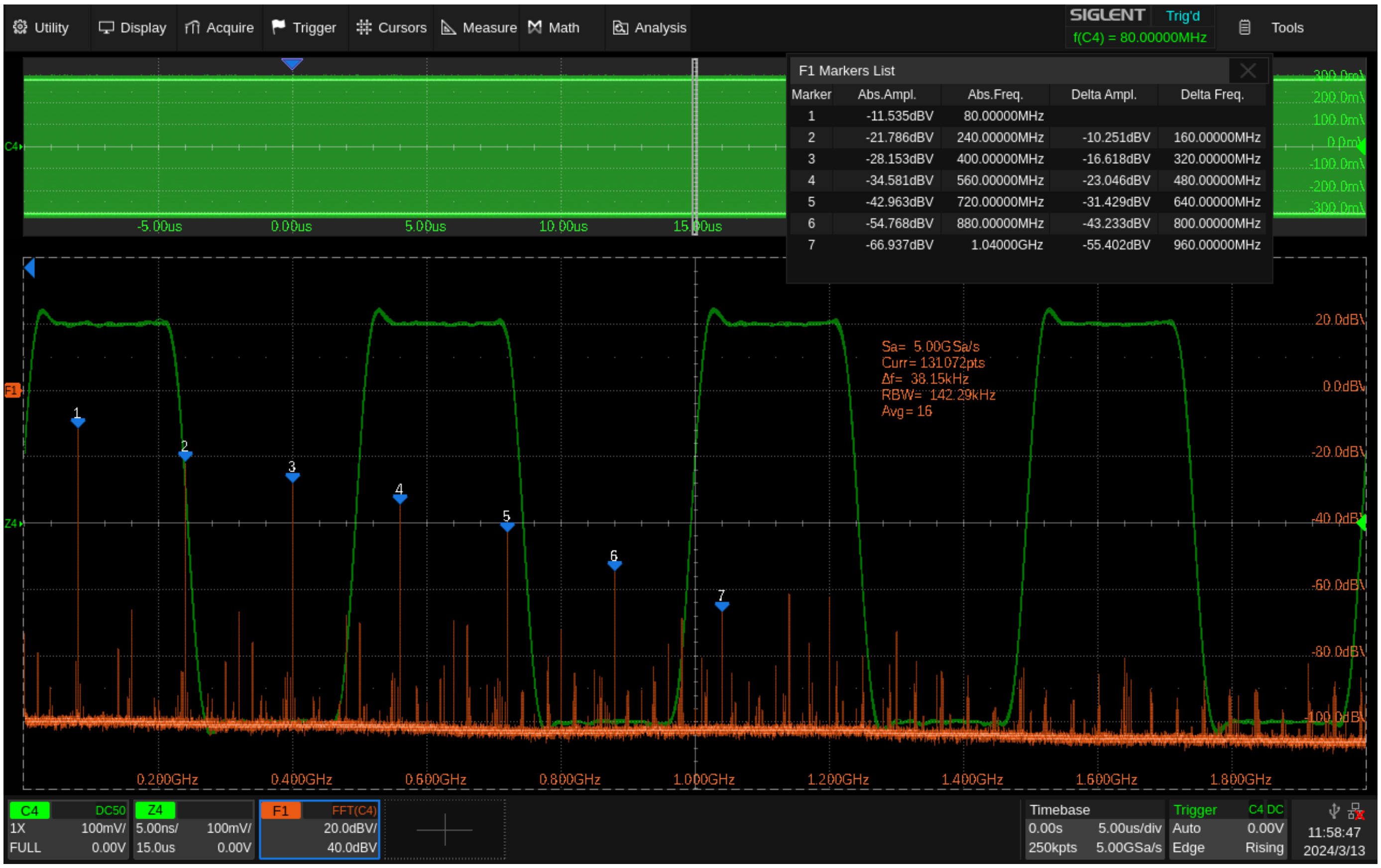

Fig. 189 Ref-Spec_Square_1ns_80MHz

Of course, it all starts with the fact that the square wave itself is far from textbook-perfect. The spectrum contains also even harmonics with a constant level of about -55 dBc, even though there shouldn't be any at all. We can also see aberrations in the time domain, most obviously a little overshoot, hence we cannot expect a good conformity with the theoretical values.

The Fourier series for a rectangular wave is like

$2 * A / \pi * (h1 + h3 / 3 + h5 / 5 + h7 / 7 \dots);$

In the example above, the amplitude is 0.6 V, so the term in front of the parenthesis is

$2 * 0.6 / \pi = 0.38197\ V_P;$

We should now be able to calculate the individual harmonics:

h1 = 0.38197 V_P = -11.37 dBV;
h3 = 0.38197 V_P / 3 = 0.127 V_P = -20.91 dBV;
h5 = 0.38197 V_P / 5 = 0.0764 V_P = -25.35 dBV;
h7 = 0.38197 V_P / 7 = 0.0546 V_P = -28.27 dBV;
h9 = 0.38197 V_P / 9 = 0.0424 V_P = -30.45 dBV;

If we compare this to the Peak Table in the above screenshot, we get the following list:

Freq. [MHz]	Calculated [dBV]	Measured [dBV]	Deviation [dB]
80	-11.37	-11.535	-0.165
240	-20.91	-21.786	-0.876
400	-25.35	-28.153	-2.803
560	-28.27	-34.581	-6.311
720	-30.45	-42.963	-12.513

The fundamental at 80 MHz is pretty close to the theory, also the third harmonic at 240 MHz is not too far off. Yet all the higher harmonics are increasingly attenuated. Well, no wonder – the textbook calculates the Fourier coefficients for ideal square waves with zero rise time!

What we want to do now, is not comparing a less than perfect square wave, captured with the SDS824X HD, to some textbook theory, but rather with the reference. The outcome is only all too predictable: at 80 MHz the measurement result will be similar, almost 3 dB too low at 240 MHz, and drop off pretty quickly at even higher frequencies.

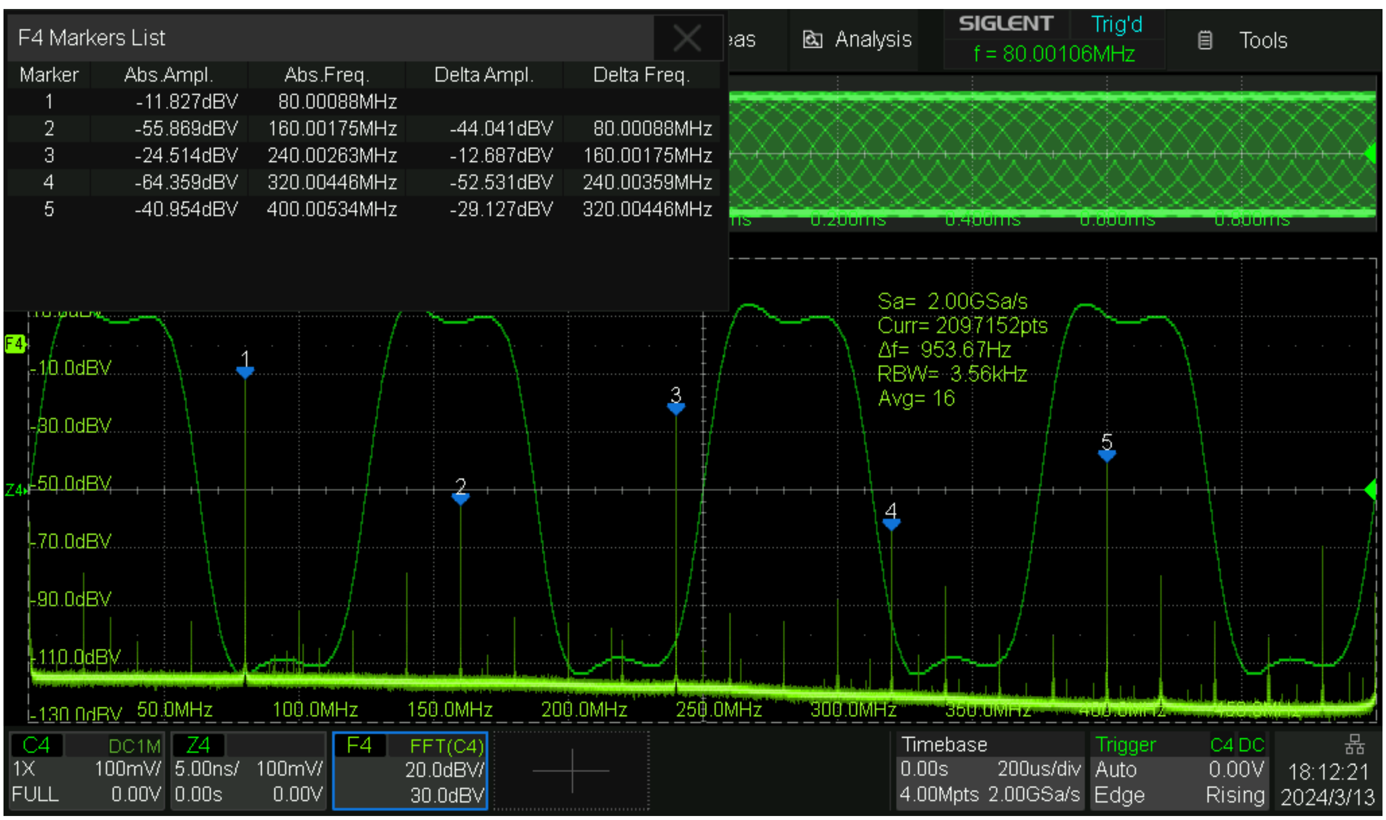

Fig. 190 SDS824X HD_Square_3.5ns_80MHz

Yes, the prediction comes true.

Freq. [MHz]	Calculated [dBV]	Measured [dBV]	Deviation [dB]
80	-11.37	-11.827	-0.457
240	-20.91	-24.514	-3.6
400	-25.35	-40.954	-15.6

Unexpectedly, the level for the even order harmonics is a little bit higher too, with -44 dBc for the 2nd harmonic.

Verdict: this is a nominal 200 MHz instrument. The true 3 dB bandwidth is somewhere at 245 MHz, as long as we don't activate more than two channels at the same time. The specified rise time is 1.8 ns, actually it is better than 1.5 ns. It has been shown that the SDS824X HD can handle pulses with 1 ns rise time, even though it cannot fully characterize them. The comparison of the Fourier series from the textbook with the real measurements was nothing more than a little fun, because in the real world a perfect square wave doesn't exist.

And the most important part: the frequency of a square wave is only important because it also dictates the maximum rise time of the signal. For instance, we cannot have a 200 MHz square wave with just 3.5 ns rise time. Other than that, especially for digital communications, we don't need excessive bandwidth – just enough to capture the relevant part of the modulation spectrum, which has to be bandwidth limited at the transmitter side anyway. Sections "SPI Speed Test" and "The 200 Mbps SPI challenge" deal with fast digital signals, and as the title already reveals, it is possible to handle 200 Mbps communication with the SDS824X HD – just 245 MHz bandwidth is enough for that.

Comment by BRZ.tech

In message #175 I asked if you accept "beginners" asking you questions about "obvious things".

In your message #180 you replied: "don't mind people asking questions here, even newbie questions 😉"

Yes, you are right. I might have just been a bit surprised. Since this is a "Review & Demonstration" thread for a specific digital oscilloscope, I expected a newbie-question in this context to be from someone unfamiliar with Siglent scopes – or even digital scopes in general, so I was prepared to deal with DSO specifics, but not with basics of signal theory and Fourier series for imaginary signals that can never exist in the real world.

Comment by BRZ.tech

On YouTube, on the R&S channel and on the Teledyne LeCroy channel, there are "theoretical videos" on the subject of "square wave excitation frequency response", in a different way from your presentation. They just don't have the practical part. In summary, they state that the BW of the DSO must be greater than 5x the frequency of the fundamental wave signal.

Well, I don't think much of such videos. Firstly, we need not tell serious users of an oscilloscope what bandwidth they require for displaying a square wave, and even more importantly, there's no use in telling some street number like "5 times the square wave frequency". Because it is simply not correct.

Comment by BRZ.tech

2. In my opinion as a "beginner", for a non-SIGLENT user (not yet), the fact that it presents the "Bandwidth" of a sinusoidal signal, and "Pulse Response" does not allow us to conclude that the frequency is tolerable with deformations, for the square wave SDS800X, is f=80MHz (photo: SDS824X HD_Square_3.5ns_80MHz).

Based on the "Bandwidth" of sine signal, and "Pulse Response", how did you reach the conclusion of BW for square wave?

Rise time is the key and I've demonstrated it. Here is how I would deal with it on a theoretical basis:

We were talking about the SDS824X HD, an oscilloscope whose bandwidth has been measured to be about 245 MHz. There is the well-known formula to estimate rise time from bandwidth:

tr = 0.35 / 245 MHz = 1.43 ns.

The SDS824X HD is specified for 1.8 ns (which it probably has if more than two channels are active), yet with just a single channel, I had estimated its rise time from my measurements as ~1.5 ns and the formula tells us that it might actually be even a tad lower than that. For the ease of use and the sake of this exercise, let's just continue working with the initial assumption of 1.5 ns.

We can calculate the total rise time tr from the signal rise time trs and the scope frontend rise time trf:

$tr = \sqrt{(trs^2 + trf^2)}$;

The error would then be: $err = (tr / trs - 1) * 100$ [%];

Rewriting this formula to get the permissible signal rise time for a given error margin leaves us with:

$trs = \sqrt{(trf^2 / ((err / 100 + 1)^2 - 1))}$;

How big an error are we willing to accept?

1%? Then the signal rise time should not be faster than 10.6 ns.
2%? Then the signal rise time should not be faster than 7.5 ns.
5%? Then the signal rise time should not be faster than 4.7 ns.
10%? Then the signal rise time should not be faster than 3.3 ns.
15%? Then the signal rise time should not be faster than 2.65 ns.
20%? Then the signal rise time should not be faster than 2.26 ns.

Or we don't care? Then everything is fine, as long as at least the fundamental frequency does not exceed the bandwidth of the DSO.

I've always used the spectrum of a signal to decide how much bandwidth is needed to get a faithful reproduction on the DSO, just as I've demonstrated earlier with the 3.5 ns rise time square wave. Once again, it's up to the user to decide what level of harmonics needs to be within the acquisition bandwidth. And of course we cannot accurately determine the expected error from this, but with some imagination and experience, we can know what we really need, also depending on the task on hand.

I have already pointed out that in digital communications we never had any bandwidth to waste, so all the digital modulations always really transported more or less distorted sine waves, which only get converted back into square waves within the receiver.

Comment by BRZ.tech

3. As you demonstrated that for the SDS800X, BW = 245MHz for sinusoidal signal, I just divided it by 5, and asked you to start the test at f=50MHz, and increase the frequency, and you arrived at f=80MHz with a "square wave" ("photo: Ref-Spec_Square _1ns_80MHz"), which 80 x 3 = 240MHz…

I've shown the two extremes. First, the benign and still fairly accurate 3.5 ns, where we can expect an error of max. 8.8%. Second, aggressive 1 ns rise time with up to 80% error. The actual errors were lower, partly because the rise time of the SDS824X HD is a little faster than the 1.5 ns that we've assumed here. Anyway, we could define a "square-wave-bandwidth" only if we have a clear error margin in mind. The 80 MHz square wave with 3.5 ns rise time was rendered perfectly okay by the SDS824X HD, even though it looked pretty much like a sine. Yes, there are irritations, because some folks probably expect to get a trapezoid waveform instead of a sine, and that is actually where the bandwidth limited square wave collides with our imagination. But how do you define aberrations in the expected shape of a signal edge? As a consequence, I would say the signal rise time is the more readily available and less complicated metric.

Comment by BRZ.tech

4. As for the video by "Professor Michel van Biezen", for experts, the issue comes down to just the photo at the beginning of the video… the equation and the Fourier Series Coefficients.

In my "beginner" analysis, it doesn't matter much if the theoretical equation doesn't support a square wave signal of 1ns risetime…

I have to admit that I have difficulties understanding what you actually want. In a previous post you wanted me to show how the Fourier coefficients of the DSO measurements match the theory shown in that video, now you claim that accuracy doesn't matter for you.

Comment by BRZ.tech

5. As for the Fourier Series, you learned another formula that I had never seen. This Fourier Series formula has many ways to learn it.

Well, in practice me at least do not deal with instantaneous time-domain values like cos(ω * t), but with harmonic signal levels from the frequency domain.

Comment by BRZ.tech

Having observed your comments, it is still not possible to understand whether the "Fourier Series Coefficients" should be placed in V_P, V_{PP}, Vrm, or something else.

In my example it has quite obviously been a mixture of all. The formula, which originally includes 4 times the peak value (which is actually half the amplitude) as first term, has been simplified by me by using double the amplitude instead. It goes without saying that the amplitude of an ideal square wave is identical with its peak-to-peak value, hence V_{PP}. The Fourier coefficients are still related to half the signal amplitude; hence they are peak values, which I have clearly stated in my article (V_P). Finally, I have converted the peak values to dBV_{RMS}. Yes, I've just written dBV, because dBV_{RMS} happens to be the only dBV unit available in Siglent DSOs (and many others).

Comment by BRZ.tech

Request: Even if it contains errors greater than 20%, as there is no perfect square wave, if you can assemble and present an equation of the Fourier Series, with f=80MHz, of the square wave, I will be very grateful. As in the formula presented in the video by "Professor Michel van Biezen".

Sorry, I honestly don't know what you want. The equation for a Fourier series looks always the same, no matter if it is for 1 kHz or 80 MHz.

Maybe someone else can chime in, because I personally am at my wit's end. All I can do now is strongly recommend that you stop fixating on that video.

Comment by BRZ.tech

6. In the photo: "Ref-Spec_Square _1ns_80MHz", in the "Peak Search Table", in "Marker 6" it has f=880.00000MHz. I said here that it has "8 digits", but you *repeatedly state that the SDS800X has "7 digits" in the Counter. Do you count from "0 to 7", or from "1 to 8"?*

Yes, the math might show more digits in order to provide more resolution for detailed analysis, but that is not necessarily accurate in absolute terms.

What I am referring to is the always visible trigger frequency counter in the top right corner of the screen. This is the closest thing to a real frequency counter and it has class-leading 7-digits resolution, albeit only pretty average 25 ppm absolute accuracy.

Comment by BRZ.tech

7. After your "class", I agree with you that the "Risetime" is the starting point for buying an AWG, and "the good one is the 1ns", but it is the top of the line, and is far above of my hobby budget.

Once again, I'm not sure to understand you. While it is obvious that it's quite nice to have a professional grade AWG that can deliver fast rise time signals, this is a completely different topic. I was always trying to get the message through, that a square wave should be primarily judged by its rise time and not frequency, even though – and this once again is something I've already stated – the two are somewhat related. A square wave at 240 MHz just cannot have a rise time significantly slower than 1 ns.

The actual message is: on a SDS824X HD you can faithfully reproduce a 50 MHz square wave with 3.5 ns rise time, while you are not able to do the same with a 1 kHz square wave that has 1 ns rise time - even though the scope-bandwidth is 245000 times the square wave frequency!

Pulse Fidelity

Comment by Orange:

I'm not impressed by the HF response step size the first 15ns; there is much undefined over and undershoot. The same behavior can also be seen on the SDS1000HD. Almost as if they use the same front-end design.

I've demonstrated how the details of a pulse flat can be closely inspected without overloading the scope input by means of a proper zoom implementation and additional math to deal with noise. Yes, neither pulse generators nor oscilloscope frontends are 100% ideal, especially in terms of impedance matching. Because what we see in my screenshots are clearly reflections stemming from less than perfect impedance matching. On the other hand, small aberrations in the realm of 2.5 % (as in this case) are usually well accepted in the industry.

I’ve added a paragraph and a screenshot to the “Pulse Response” section to demonstrate a modified test setup for the occasions where we really need highest possible signal fidelity up to the fine details.

Comment by Orange:

Can you also make a test on a 100Hz square wave input signal? Some recent Siglent scopes also have problems with this (not showing a real square)

Absolutely no problem with a 100 Hz square when using the SDS800X HD (or any other contemporary Siglent DSO):

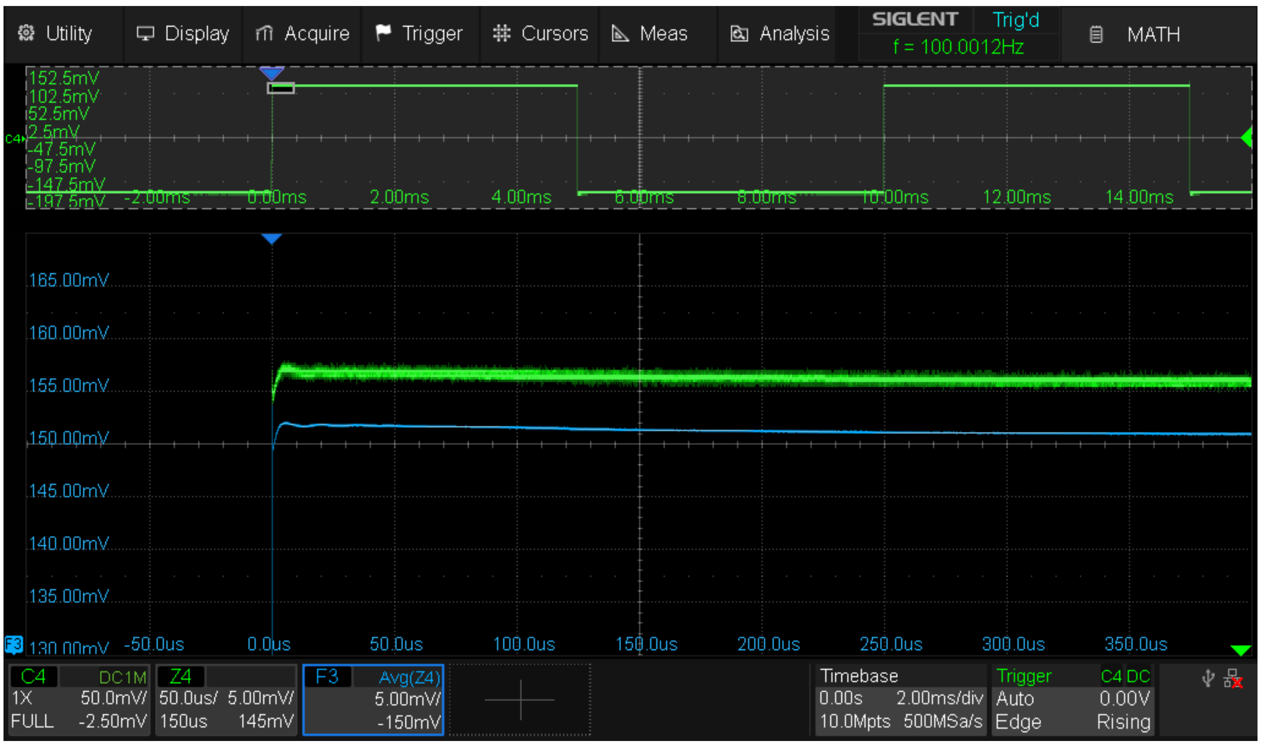

Fig. 191 SDS824X_HD_PR_100Hz_Zoom

Also, claims about serious problems with the SDS1000X-E (at 10 Hz this time) could not be confirmed:

https://www.eevblog.com/forum/testgear/siglent-sds1104x-e-in-depth-review/msg5278090/#msg5278090

Outstanding Features

Comment by Tszaboo

Any suggestions what to test that's missing from a MSOX3104, what I used at my different jobs for the past 14 years or so?

I have great sympathy for people who are just content with what they have – and only having what they really need. Requirements for T&M gear in general (and DSOs in particular) can vary widely, depending on the tasks on hand. And of course, why should an instrument that met your requirements 14 years ago not still be the tool of choice today, as long as your requirements and expectations haven't changed during all these years?

Yet it sometimes cannot hurt to reconsider the requirements, otherwise one might miss some opportunities to make life easier. There are lots of examples in this book of what even a lowest end analytical scope can do nowadays:

What about 12-bit resolution and a nice vertical zoom implementation as demonstrated in the chapters "Pulse Response", "Vertical Zoom Demo", "Zoom Expectations" and others?

And wouldn't it be nice sometimes to have true 500 µV/div full resolution sensitivity, instead of just 4 mV/div and everything below that just fake, as demonstrated in "True Vertical Sensitivity"?

Could it help sometimes to have a low noise instrument with <2.4 nV/√Hz voltage noise when 50 Ωterminated as demonstrated in the "Noise" chapters?

Is History something useful, where you can find up to 80000 past trigger events (records), play them back and analyze them to your heart's content using all the tools the scope has to offer? Look at "History & Sequence Mode".

What about deep measurements, which have deep memory as a prerequisite? Sections "Deep Measurements" and "Counting Pulses" give some examples of tasks that cannot be completed with short memory and measurements on heavily decimated data.

Aren't the small Histicons in the measurement statistics convenient, as demonstrated in chapter "Measurement Histograms" and many other sections where automatic measurements are used?

You might value measurement Trend plots at times – wouldn't Track plots be very useful as well, as they allow you to clearly visualize frequency and phase modulated signals as demonstrated in "Track Plots"?

Wouldn't it sometimes be nice to be able to make accurate measurements? For very low frequencies and DC we can pull out a (good) DMM, but is it really necessary if you can have a DSO with 0.5% DC-accuracy, as confirmed in "DC Check"?

When I still worked with analog CROs, I made sure to have one with trigger signal output, where I could connect an 8-digit frequency counter, so that I've always been able to know the exact signal frequency, as it was important for many of my tasks. That's become a habit and since trigger signal outputs were never very common and completely went out of fashion at one point, I wouldn't accept any scope without a permanently visible precision frequency counter, like the 7 digits in any contemporary Siglent. Also mentioned in the "Counter"-section and numerous examples can be found throughout this Review.

4 highly sophisticated math channels with formula editor allow various experiments as well as extensive data conditioning (not only) for the FFT.

A deep (2 Mpts) FFT in turn allows good frequency resolution, and it's all the more useful if we can have four FFTs at the same time as demonstrated in some of the noise measurements. A high dynamic range of up to 100 dB as demonstrated in "FFT Dynamic range" should also be welcome.

What about a powerful Frequency Response Analysis (Bode Plot) with up to 3 channels up to 120 MHz and >100 dB dynamic range, as demonstrated in "Bode Plot at a glance" and "Bode Plot Example"? That's for free, by the way, not part of an expensive Power Analysis package.

Maybe the two custom probe definitions would be more convenient than a bunch or predefined probe factors, see "Custom Probe Factors"?

Pulse Fidelity versus Anti-Aliasing

Comment by EvgenyG

Looks like there's a bit of overshoot when in 824X mode compared to 804X. I wonder if the real 824X would show same behaviour.

I'd rather have correct signal display at 70Mhz bandwidth, than observing overshooting that is not there.

It is just not possible to meet everyone's taste. There is one fraction of people, who thinks Siglent "does it right this time" by fitting a "proper" anti-aliasing filter in the frontend, where "proper" usually means "effective", like in reply #417 here:

https://www.eevblog.com/forum/testgear/siglent-sds3000x-hd-and-upgraded-sds1000x-hd/msg5376662/#msg5376662

Then there is another fraction, complaining whenever the slightest pulse distortions (in form of overshoot) become visible.

Anyone who has ever tried to design a "proper" AA-Filter (and thoroughly analyzed the outcome) will have learned that this is just not possible, without severely compromising the pulse fidelity, for a number of reasons.

The only near perfect solution would be to use massive oversampling, like in the ancient boat anchors, that didn't have sufficient processing power to apply a true sin(x)/x reconstruction, hence had to make do with simple linear interpolation, which in turn only works as long as the sample rate is at least ten times the signal bandwidth.

In the latter scenario, we could use a 7th order Bessel filter to get 60 dB attenuation at the Nyquist frequency – or a 3rd order Bessel to get about -34 dB together with a digital FIR filter to provide additional attenuation in the 1st Nyquist zone, i.e. from the rated bandwidth up to Nyquist, which would then be about five times that bandwidth.

But with that “ideal” approach, another (huge) fraction of people would pop up and loudly complain about a scope with only max. 50 MHz bandwidth (and 4 x 500 MSa/s), or alternatively, a 200 MHz scope that’s sooo expensive (because it has to have 4 x 2 GSa/s).

That’s not a Siglent-specific issue; all the top models of a product line of any serious manufacturer have to make the same tradeoff, either strong aliasing or bad pulse distortions. Yet this problem seems to hit Siglent in particular, because their products are still affordable and, even more importantly, easily hackable (at least the low-end devices) and everyone who takes pride in themselves feels inclined to hack it for the highest possible bandwidth.

The SDS 800X HD has a moderate(?) AA-filter in the frontend, presumably it’s a 3rd order Butterworth (I’m just speculating here), which of course causes significant overshoot. It’s even specified in the data sheet: 10% with a 150 ps rise time pulse from a 50 Ω source. Not many manufacturers are brave enough to even specify that in their data sheets!

When more than 2 channels are activated and the sample rate drops to just 500 MSa/s, an additional digital filter is activated to limit the bandwidth to true 200 MHz (~245 MHz otherwise) and aid aliasing suppression in the first Nyquist zone (200-250 MHz).

The lower bandwidth models just activate the most appropriate bandwidth limit in the integrated PGA (Programmable Gain Amplifier). The bandwidth limit has to be much higher than the specified bandwidth, because there is a limited number of choices and the corner frequency is subject to high tolerances of ±20% or even more (because we cannot have accurate resistors on a silicon chip) and we don’t want to risk any number of individual scope samples to not meet their specifications.

This means, we will usually measure a bandwidth significantly higher than specified, but it will vary from sample to sample and we cannot rely on just a few individual reports.

In any case, this integrated bandwidth limiter is just first order, hence will have Gaussian response and not cause any pulse distortion.

What has been stated above is also true for the switchable 20 MHz bandwidth limiter.

Figures

Tables