The Embedded Muse 45

Delayed Sweep

I had the chance to look at another new scope that will soon hit the market, one aimed at the mid-level developer and priced accordingly. It's a very nice piece of gear. However, for some reason the machine lacks delayed sweep. The vendor tells me that few engineers use this feature.

What a loss! I find delayed sweep almost essential for working on fast digital signals. Here's an overview:

Just as any decent scope has at least two vertical channels, most include two time bases as well. Seems odd, doesn't it? Double vertical channels intuitively makes sense, since each probe picks off a different sense point. Time, though, always flows in the same direction at the same rate, so a single axis is all that makes sense.

Novice scope users understand the operation of time base A: crank the time/division knob to the right and the signal on the screen expands in size. Rotate it to the left and the signal shrinks, but much more history (i.e., more microseconds of data) appears.

Time base B is a bit more mysterious. If enabled, it doesn't start until sometime after time base A begins. Try it on your scope: select "Both" (or "A intensified by B") and select a sweep rate faster than that used by A. You'll see a highlighted section of the trace whose width is determined by B's sweep rate, and whose starting position is a function of the delay time knob.

Switching from "Both" to "B" shows just the intensified part of the sweep: the part controlled by time base B. In effect, you've picked out and blown up a portion of the normal sweep. It's like a zoom control - and you can select the zoom factor using the sweep time, and the "pan position", or starting location, using the delay time adjustment.

Suppose you want to look at something that occurs a long time after a trigger event. Using these zoom controls you can get a very high resolution view of that event - even when time base A is set to a very slow rate.

Delayed sweep is always accompanied by a second trigger system. Any instrument with dual time bases will come with a second of these knobs to set the trigger point of the B channel.

The second trigger is important when working on digital signals that usually have unstable time relationships. Set the A trigger to start the sweep (as always), position the intensified part of the sweep to some point before the section you'd like to zoom on, and then adjust trigger B until the bright portion starts exactly on the event of interest.

This procedure guarantees that even though the second trigger event moves around with relationship to trigger A, you'll see a stable scope display after selecting the B time base. In effect you've qualified trigger B by trigger A, hopefully zeroing in on the area needing study

Suppose your microprocessor crashes immediately after RESET. Traditional troubleshooting techniques call for hooking up the logic analyzer and laboriously examining all of the data and address lines. Personally, I find this to be too much trouble. Worse yet, it tends to obscure "electrical" problems: the analyzer might translate marginal ones and zeroes into what look like legal digital levels. Logic analyzers are great for purely digital problems, but any problem at power-up can easily be related to signal levels.

Only a scope gives you a view of those crucial signal levels that can cause so much trouble. Trigger channel 1 on the RESET input and probe around with channel 2. Look at READ: every processor starts off with a read cycle to grab the first instruction or startup vector. You may find a puzzling phenomena: if the reset is provided by a source asynchronous to the processor's clock (as is the case with an RC circuit, a Vcc clamp, and even with many watchdog timers) READ will bounce around with respect to RESET. You'll never get a nice high resolution view of READ this way.

Triggering off READ will not help. You need to catch the FIRST read after reset (to look at the first instruction fetch), not any arbitrary read... and no doubt there will be millions of reads between resets.

The answer is delayed sweep. Put RESET into the scope's external trigger input and fiddle the knobs till you get a stable trigger. I like to put one scope channel on the external trigger while doing this initial setup to make sure the trigger is doing what I expect. Then, connect channel 1 to your processor's READ output and crank the time base till it appears over towards the right side of the display. Go to delayed (A intensified by B) mode, and rotate the B time base trigger adjustment until the bright part of the trace starts on the leading edge of the bouncing READ signal.

At this point time base A starts the sweep going on the asynchronous RESET, and time base B triggers the intensified part of the sweep when the first READ comes along. Flip the Horizontal Mode switch to B (to show only the intensified part of the sweep - that part after the B trigger), and a jitter-free READ will be on the left part of the screen. Cool, huh?

With the now stabilized scope display you can use channel 2 to look at the data lines, ROM chip selects, and other signals during the read cycle. It becomes a simple matter to see if the first instruction gets fetched correctly. A lot has to be perfect for this to happen. Very often a power-up problem comes from a bad data line, chip select or buffer problem, any of which is trivial to find with the scope triggered properly.

Delayed sweep was available way back in the vacuum tube scopes of Tektronix's 540-series; it's a shame to see it forsaken today.

Thought for the Week

```
I keep hearing that nobody cool uses goto's in C.
To which I say:

    C'mon, give me a break;
        else,
            I'll continue;
```

About The Embedded Muse

The Embedded Muse is an occasional newsletter sent via email by Jack Ganssle. Send complaints, comments, and contributions to him at jack@ganssle.com.

To subscribe, send a message to majordomo@ganssle.com, with the words "subscribe embedded *your-email-address*" in the body. To unsubscribe, change the message to "unsubscribe embedded *your-email-address*".

The Embedded Muse is supported by The Ganssle Group, whose mission is to help embedded folks get better products to market faster. We offer seminars at your site offering hard-hitting ideas - and action - you can take now to **improve firmware quality and decrease development time**. Contact us at info@ganssle.com for more information.

The Ganssle Group, www.ganssle.com