From: Bob Jacobsen <Bob_Jacobsen@lbl.gov>
Subject: **Comment on technical basis of modified S9.x.1**
Date: February 16, 2011 10:53:59 AM PST
To: Didrik Voss <davoss@pvmtengr.com>

In this note, I'm not going to address concerns about process, policy or how recent actions will effect volunteer efforts. Those are real concerns, which need to be resolved, but here I want to focus on a single issue: The technical reasons behind my statement that your modified S9.x.1 cannot live up to the promises that it makes. I've made every point in here to you before. I'm putting them all in this one note to make it easier for you to provide it to the Board next week.

Di, as a licensed professional engineer, I'm sure you can appreciate that creating a robust standard involves more than just a couple of limiting calculations. When you sign off on something, you're saying that it meets _all_ the constraints, with sufficient margin that it will reliably function under _all_ of the required circumstances it will encounter. It's not just about meeting one or two criteria. It's about putting calculation and experience together to create a robust specification.

I too was trained & practiced as an engineer, and I care very much about engineering statements that I make.

In my considered opinion, based on both experience and calculation, equipment built to your modified S9.x.1 will _not_ function reliably in the entire region described.

Both experience and calculation lead me to this conclusion. Let me start with experience. I've been working with field bus installations since 1977. I've been extensively using CAN in laboratory and shop areas for at least 10 years. But I want to focus on results from two organizations, not just individuals, with relevant experience.

The first is Rockwell International, which makes industrial automation equipment based on CAN. This is a very professional company that builds just about everything rock-solid. They decided that they would design a new cabling system for CAN, using another kind of cable to achieve certain commercial and performance goals. Here's a report on _some_ of what they went through. (Full disclosure: I had some peripheral involvement in this effort, but didn't contribute to these tests or this report)

http://www.odva.org/Portals/0/Library/CIPConf_AGM/ODVA_12_AGM_New_LowCost_DNet_Media_Lounsbury.pdf

That's about 20 pages (some of it is about some 250kbps & 500kbps testing, so I didn't count that) of calculations and tests to characterize their specific recommendations. This is 20ga wire, with power in the cable, both of which indicate that it should behave _better_ than your modified CAN which is 24ga wire without power in the cable. Note that this set of tests involves many more criteria than just the two static equations that go into your Figure 1. There are AC impedance calculations, effects of lumped vs distributed nodes, etc, all of which they worked through.

Because this is a different wiring system that the RJ45/UTP that NMRAnet will use, their results are not directly comparable to your figure 1. Instead, there are two things that should give you pause:

1) This is significantly more testing and analysis than the DC calculation based on two application notes that you did

2) Even with the larger wire and isolated power, they only went to 64 nodes.

The other organization is Lawrence Berkeley National Laboratory. (I have an appointment there as senior staff, in addition to my campus professorship.) LBL does fore-front research in many areas, and uses field buses in many ways. From running accelerators and X-ray light sources through small desk-top control systems, we use CAN networks in many capacities. The lab has developed guidelines beyond which we will not go without custom engineering:

* 200 m max
* 50 nodes max
* no stubs over 6m
* no cables under 0.5m

(Some minor caveats: Most LBNL work is at 250kbps, so that's where the recommendations were primarily developed; the 125kbps recommendations are the ones that matter for NMRAnet, and are the ones quoted here. These numbers are different than what's in the consensus S9.x.1, because the limits there are expressed differently and the LBNL work is usually done with heavier-guage DeviceNet cables that have better properties; the consensus S9.x.1 is based on significant engineering discussed below in addition to these guidelines.)

These are based on long experience of many people. Note that these limits are for a relatively clean lab environment, with professional quality equipment & high quality cabling, for trained engineers who are happy to pull out a $20K instrument when something needs to be done. If we're not comfortable going well beyond this, I don't think we should expect model railroaders to be.

As an appendix below, I've included a writeup that I did for the NMRAnet working group a while back, when I was asked for some examples of what can go wrong. It's part of another conversation, somewhat different from this one, so it has some non-sequitors, but it shows the kinds of things that I have personally seen happen in large CAN networks. CAN networks that exceed those recommendations can and do malfunction, and they can be very hard to diagnose and repair.

In addition to experience, engineering involves calculation and modeling. That's what I'll discuss in the rest of the note. Let me start by saying that I'm not going to quibble about the DC loading calculations you've used to make Figure 1. I have some small differences with them, particularly as expressed in the draft, but those are minor. Perhaps we can resolve them when and if an RP is written around the modified CAN and it's performance.

What matters here is not the details of the Figure 1 contours, but rather _other_, non-included effects that can cause an NMRAnet built to your modified S9.x.1 to fail in a place where modified S9.x.1 says it should work.

I'll focus on the upper-right corner in the plot, the place where maximum node count is at its maximum length.

There are two possible worst cases there: Nodes distributed evenly along the length of the network, adding fully distributed load. And nodes primarily lumped at one end, communicating with a single node at the other end. Note that the Rockwell tests checked multiple arrangements to cover cases like these; your formulae don't distinguish them, which is another indication that your formulae don't cover everything.

In the lumped-and-single case, the capacitance of the lumped nodes cannot be ignored. This is particularly bad without the minimum cable length that you removed from the consensus draft. With connectors, 100 notes is a lumped capacitance of approximately 2.5nF. Without spacing between nodes, this is essentially a single capacitive load of 30 ohms at the transition-edge frequency (2MHZ with standard CAN sampling for 125kbps; see Philips AN 97046 and references therein). That's a significant addition to the standard termination load, and combines with it and the DC receiver loads to cause reflections in the line. You can see this effect in action in some of the figures in the URL above. The effect is difficult to calculate in the abstract, because it depends on the details of each particular case, but it's the reason that the length falls off with number of nodes so much faster in the LBNL recommendations and the consensus S9.x.1 than it does in your modified S9.x.1: Shorter networks are required so that the drivers have have time to properly condition the signal.

When you removed the discussion of stubs and minimal cable length, you also removed protection against another set of time-domain problems. These are mentioned in section 2.4 of the consensus TN, along with a pointer to a TI note that covers them well: http://focus.ti.com/lit/an/slla270/slla270.pdf (I don't object to learning about limitations from an application note; I object to picking just one, and ignoring all the others that are also telling you important things that need to be considered) The issue here is that short cables and stub cables, e.g. to throttles, can also cause time-domain problems for CAN networks. Even at 125kbps, in the largest networks these can reduce one's margins to the point where the limits of your DC calculation cannot be reached.

Finally, I disagree that the provision of capacitor-coupled split termination will have a large positive effect on noise margin. That's not the origin of split termination, which instead is usually motivated by reducing electromagnetic compatibility (EMC, e.g. noise) issues where CAN _radiates_ noise that causes trouble for other equipment. See for example page 22 of Phillips AN96116, cited as such in TN9.x.1. CAN itself is quite robust to common mode noise due to it's transmission design, but there can be significant radiation from common mode signals on its cables which occur due to ground offsets. Power-in-cable (which was also removed from the draft by you) can reduce this by reducing unbalanced ground currents on long cables, as can suitable split termination. Split termination does nothing to improve differential noise, in fact in some cases it can make it worse; differential noise is what interferes with CAN itself. The consensus S 9.x.1 draft benefits from the multi-twist character of UTP cable to cancel external differential noise, which it does _extremely_ well. UTP is commonly used in much harsher environments than model railroads, to carry smaller margin, higher speed signals, and it does that successfully.

I'll conclude starting with my prior point: In my professional judgement, the modified S9.x.1 cannot reliably reach the performance levels it promises. I have sketched my reasons, but that's just the summary of the months of effort & correspondence that went into this in summer and fall 2010. I believe that you have taken my work, and the work of the others in the group, and made modifications to it so that NMRAnets will in significant cases fail to operate reliably, even though they comply with the modified S9.x.1 Standard. I do not want my name associated with that result, and I suggest that the NMRA probably doesn't either.

If in your professional opinion your modifications to CAN do rise to a sufficient level of reliability, and you're willing to author a new Standard, that's fine. You should do that and present that new Standard to the Board at a later date.

Absent that, it is my earnest hope that you will discuss these concerns with the Board, and that they will adopt the original consensus S9.x.1 and TN9.x.1.

Bob

Appendix: Discussion of prior experience, copied from NMRAnet WG post, lightly edited to remove some off-topic references.

Most of my CAN direct experience is with installing and debugging CAN networks at 250kbps, with some at 1Mbps (only when needed for speed), and a few at 125kbps. Most of that experience carries over, even from 250kbps, because when you're engineering a network, speed is one of the parameters. You learn where you can't go beyond the edge of the calculations at 250kbps and that carries over to calculations at other rates; this is called "engineering".

But I was asked for specifics at 125kbps, so I'll provide a summary. I had to go back through my notebooks to verify some details, sorry for how long that took, but here are large-scale 125kbps CAN systems that I've worked on that were near CAN limitations. (The labels won't mean much to anybody else, but will help me find the info if I have to go back to the notebooks for more) I've omitted lots of small plug&play work that doesn't really teach much about working with CAN limitations.

* BaBar lab net (50-2139) - This was 21 nodes and a total physical extent of 120 meters in an office/lab environment. It used DB9 connectors soldered on CAT3 cable (a decision that made sense at the time to save money, but we would _never_ even consider afterwards; the physical connections were awful). Nodes were arranged in 5 locations physically spaced about 2, 12, 7 and 90 meters apart. The isolated location was two PC nodes in an office. Total cable length seems to be about 160 meters. It was a linear connection with no drops. This network did not function at its designed 250kbps (it was right at the corner of propagation time and AC loading, messages from the remote PCs wouldn't arbitrate reliably) and was dropped to 125kbps were it worked well with about 900 nsec of measured time margin.

* BaBar HV II - This is the one mentioned in the discussion above. It's two clusters of nodes 50m physically (60m cable) apart, 11 nodes in one cluster, 13 in the other. The two clusters had been built by independent groups using DB9 flat-cable jumpers between nodes. When we tried to connect the clusters at the designed 250kbps, they failed arbitration tests. We dropped to 125kbps where the problem was worse because of where the reflections lined up. We were able to get it to work at 125kbps by using custom (tuned) terminators & custom node timing, but that wasn't considered an acceptable long-term solution. Steve Lewis found a paper on the minimum spacing (I think it's the referenced app note, but don't know for sure, as I didn't write down the citation). We switched to 1m (3'?) cables between the nodes (what we had on hand). It worked with minimal reflection at 125kbps, so we successfully went back to the 250kbps required rate. (The reflections in the time domain are the

same, so we were pretty sure that would work)

* BaBar lab net (50-2177) - This is a rearrangement of the first network. The extension to the remote office was gone, but the total number of nodes was raised to 39 in 3 locations spaced about 20m apart. (Most of) the existing DB9 cables were replaced with commercial DB9 controlled impedance cables (they look like DeviceNet, but I don't have a part number in my notebook). Standard 1m cables where used between new nodes; the existing DB9 cables where (mostly) 2m. Total cable length appears to be about 115m. This started at 125kbps, where it worked reliably. We kept it at 125kbps because resetting the nodes was more trouble than it was worth. We didn't make any margin measurements on it. A couple months later, we tried to add 8 more nodes in the center rack (total of 47 on the network) and the two end racks started throwing errors. Chris LeClerc removed two unneeded nodes from each end rack (ending up with 6, 28, 9) and the error rate became OK. Later, margin tests were done preparatory to an expansion, and it was determined that the problem had been AC load in the end racks. (We never actually did the expansion)

* XE pulser test setup (70A) - The most recent debacle, this was an ad-hoc network that had worked reliably at 125kbps (rate chosen because of the equipment we assembled it from) as it slowly grew well beyond original expectations. It had never really been engineered or studied in detail. Suddenly (over a couple days), it started failing to arbitrate. At the time it started having trouble, it was four racks about 3m apart (5m cables) with 16, 2, 21, 21 (mostly commercial) nodes arranged in a square around some apparatus. Standard cable length was 1m, a mix of DeviceNet DB9 cables, flat cables and RJ45/UTP. After much thrashing around trying to find the wrong things (we thought a cable or node had failed, because it was such a sudden problem), it was learned that a student-with-initiative had decided to rearrange the connections between the racks from A <-> B <-> C <-> D to D <-> A <-> B <-> C, essentially going around a different three sides of the square, so that access with a portable crane to some of the diagnostic ports would be easier (the cable trays are attached to the racks about 7' off the ground, and she wanted to remove one of them). To put it another way, when the nodes were connected as 16, 2, 21, 21 it didn't work, and when they were 2, 21, 21, 16 it did. That's the point at which we stopped looking for a specific failure, e.g. cable or bad connection, and started looking at the system as a whole. It had 60 nodes arranged linearly and a total cable length of 71m, no stubs. A scope indicated the problem was (most likely) DC margin, which had been out-of-spec already before the rearrangement. A colleague installed IXAAT repeaters in the two racks with the most nodes (expensive, but he had them already) and the problem was solved.

A final note: It's true that much of my experience is with non-RJ45/UTP cables, e.g. DeviceNet, but that part of the experience is being applied to the _electrical_ issues involved here with engineered expectations for lengths and number of nodes. The argument for RJ45/UTP is only partly, not entirely, an electrical one. The electrical argument is that you want to specify a standard in terms of cables with reliable properties, so that you can reliably predict the expected performance. My experience listed above is directly applicable to that, even though not all of it is with RJ45/UTP cable. RJ45/UTP is a way to get cables with reliable properties, but it's not the only way to do that. DeviceNet cables, etc, also have reliable properties. The non-electrical argument for RJ45/UTP is that the cables are commonly available, are easy to use, and are cheaper than the other alternative cabling systems.

--
Bob Jacobsen, LBNL
Bob_Jacobsen@lbl.gov +1-510-486-7355 fax +1-510-643-8497 AIM, Skype JacobsenRG