GPS software we are working on at the moment has come from the project. <a waiter refill my water glass, this is classy>.

So, you could say that there are two main streams in your computer music - performance poetry and data processing?

Sure, you know Voyager has redefined our experiential mind. Attempts to comprehend the distance it has travelled fails in value to any measurement on earth. We have created machines that advance our knowledge but we have failed to sculpture minds that appreciate the depths of new existence. I am still met with the approach of ‘nobody will bother understanding it, you should make it accessible’, as if there are no challenges. Sol’s Violin will only benefit those with a inquisitive mind and I tend to think that they are the only people I am interested in. I wanted to mimic the process that I arrived at over three years, that you look at all this data and finally it all connects in. Ultimately I am faced with the divide of being an entertainer or an artist, whether I am a musician or a composer. <The second bottle of wine kicks in and I think we are both on our way to oblivion, full steam ahead!>. The poetry that I wanted to create was multi-voiced and I thought of computers before choirs. With the processing data either you can look at it or hear it.

Your recent comments about acoustic space on aus_noise where interesting. I gathered that you where comparing architectural space, such as the hall way of a building with that of an electronic architecture of a micro-chip or the design of a computer program? That sprung out of a conversation about ‘plugin spotting’ and how a computer program has a certain ‘sound’...

[the conversation was interrupted by knocking the water glass over the minidisc, after a small commotion we decide to have coffee and talk about ‘other things’]

Using Test Harnesses for Debugging

Angelo Fraietta

Some of you who attended the ACMC in Perth this year might remember me franticly trying to work out why Anne Norman’s bell mechanism was not working with my Smart Controller. The bell hardware was constructed by Anne in Melbourne, and I had not even seen it until the day before. I designed and built the Smart Controller unit for her as a rack mount system, which was then connected to her bell system using a pair DB 25 plugs and cables. In order to try and fix the problem, I had to determine where the problem was. I was unable to determine what was wrong during that session. That night, I made a test harness that would enable me to test each portion of the circuit quickly and accurately. The next day, although I was unable to get the bell working at first, aided with the test harness, I was able to ascertain that the power output from one socket on the power board was not working. I swapped the plug pack to a different socket, and “hey presto!” it worked.

So what is a test harness? A test harness is an adapter cable or diagnostic tools that will enable you test each part of a circuit without having to use a soldering iron. A similar thing in the PA world is the CD player and headphones. Let’s take the example that we are trying to get a PA working. You turn everything on, talk into the microphone and there is no sound. The first step you might take would be to substitute the microphone with the CD player as the audio source and commence debugging from there, checking for signal in the path. You can check for levels at various modules, such as mixer LEDs, headphone outputs, effects level displays, amplifier level displays, and the likes. These devices already have inbuilt input and output points that can be used for diagnosing problems.

That is fine for an audio setup, but what if you have a set of sensors connected up to a MIDI to CV converter going into MAX. It’s half an hour before the performance and something is just not working. The first tool you require is a MAX (or Algorithmic Composer) patch specifically designed to test all the inputs that you are using. When testing my Dumb Controllers before sending them out into the wide world, I test the CV inputs with the following Algorithmic Composer patch. (see figure 1)

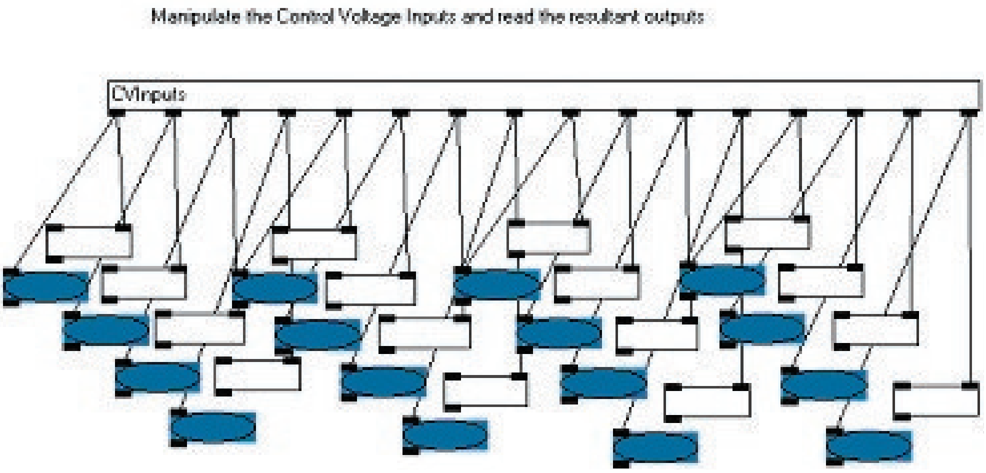

Figure 1

Here, without having to think too much, I can see which inputs are working, which are noisy, and which are not working. I also have similar patches for testing the digital inputs, digital outputs, and CV outputs. If everything is fine here, you know your problem is not at the sensors.

If you have determined that the problem is that some MIDI information is not getting to the computer, what do you do then? If a single sensor is not working, you might be tempted to swap it with a sensor from another input to see if that is the problem. I have found that this technique, particularly if there are a lot of sensors, can cause more problems – the biggest being that the sensors don't end up being in the right place after you have finished debugging. I have found that pulling the suspect sensor out and using a sensor made from 1K potentiometer in its place (for systems like the I-Cube and my Dumb Controllers) is a very quick way of testing a channel. You can make one of these up for about three dollars.

In the case of the garden bell, a different test harness was required. The way that the device operated was by removing infrared light from an IR receiver (effectively causing a high impedance), which was the input to the Smart Controller. The Smart Controller, after receiving this input, generated a 5V pulse at the mapped output, switching a reed relay on for that pulse duration, which in turn switches 12V to the solenoid, causing it to strike the bell. There are three areas to test – the IR receiver, the 5V output pulse, and the 12V to the solenoid. Examining the circuit connecting the bell clusters to the Smart Controller we have a DB25 cable, through IDC ribbon, to the bells. (see figure 2)

Each bell cluster (green box in above circuit) contains the following circuit. (see figure 3)

The IR receiver can be simulated by shorting its connections together, whereas the 5V pulse that energises the relay can be easily tested by

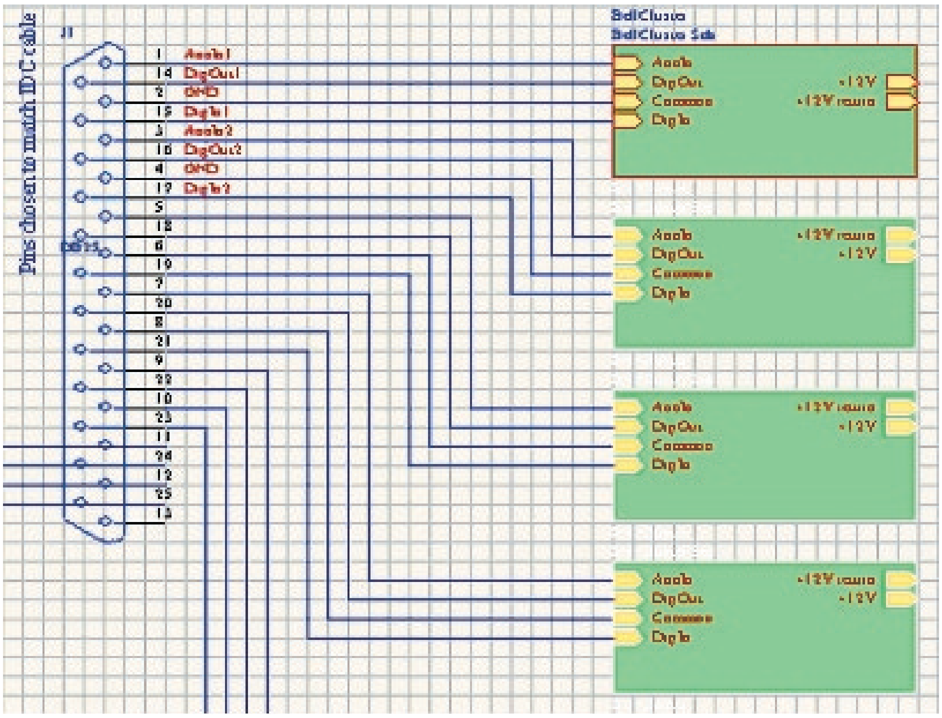

Figure 2

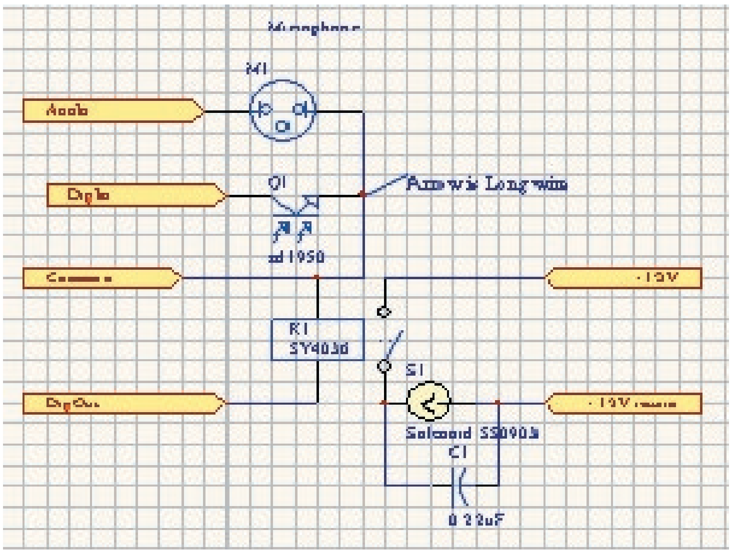

Figure 3

placing a red LED across it. A simple harness can be created by crimping a 26 way IDC line socket, which costs less than two dollars, across the IDC ribbon cable. The inputs and outputs of the Smart Controller can be easily tested by placing a LED between pins 2 and 3 to test the 5V pulse, and by momentarily shorting pins 3 and 4 to simulate the IR. The LED and shorting wire simply push into the IDC line socket, so there is no need to solder. If you momentarily short pins 3 & 4, the LED will pulse. A circuit showing this is below. (see figure 4)

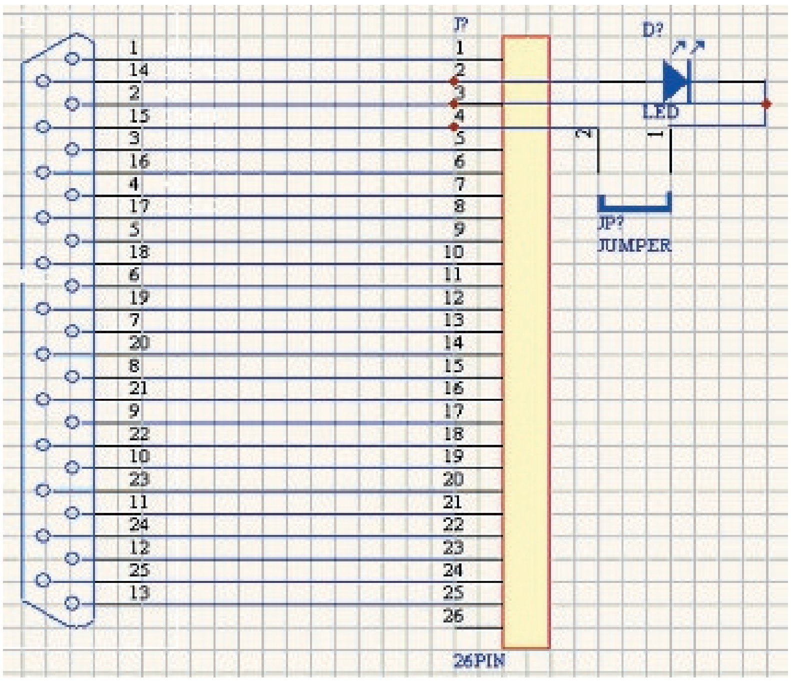

Figure 4

Each cluster can be tested by moving along four, e.g. to test cluster 2, place the LED on pins 6 & 7 and short pins 7 & 8. The added bonus of this harness is that audio triggering for Bell Garden 2 can be tested by placing a microphone into pins 1 & 2.

There are other techniques that can be used, however, the important thing to keep in mind is to keep away from the soldering iron until you absolutely have to solder. T est harnesses are easy to make and are worth spending the extra twenty minutes or so required to make them. Good luck.