

A Challenge: Define an experiment that will stump SCAT.

A year's subscription to this journal says you can't.

Now that may seem like quite a rash commitment. But then again, SCAT is quite a system — so much so that we feel the odds are really stacked in our favor. Just to let you know what you're up against:

First of all, you're up against the conversational SCAT language. Which means that we won't have to waste time programming your experiment requirements in machine language. All we do is describe your experiment through the system input teletype. It's as easy as saying, "S1, S2, S3."

Second, you're up against a very generous control and monitoring system. SCAT won't flinch at controlling 12 discrete stimulus devices (house lights, feeders, solenoids, valves, etc.) and at monitoring 10 different digital response inputs (from levers, keys, bars, photocells, etc.) *per station*.

Then, there's what we call the data elements: Clocks, Constants and Counters. If you beat the system, you will have to use over 10 different elapsed-time conditions and 20 different counting parameters.

You say you may want to change experiment parameters frequently? No sweat. Data elements can be queried at any time, and their contents changed through the control teletype.

Next, you should be aware that SCAT can add, subtract, multiply and divide, and do it all in decimal.

It can also make comparisons, then branch in various directions depending on the relationships (greater than, equal to, less than, etc.) it finds. So you might as well forget relying on number manipulations to put SCAT out of the running.

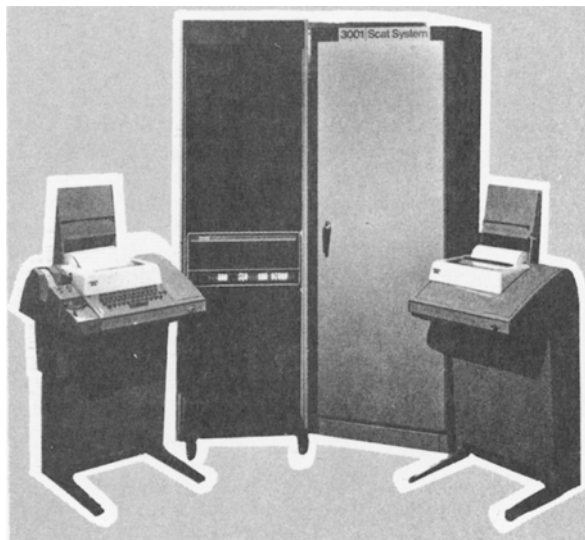
We should also warn you that the system is event-driven rather than time-driven. So don't suppose that your stipulation of real-time control and data acquisition will slow us down. SCAT recognizes each event as it occurs, to assure that no data will be overlooked or lost. That, among other things, is what the interface hardware is all about.

Now, about data. We've already mentioned the SCAT arithmetic facility that lets you condense data. There's also List (that will type out sequentially whatever data you choose to store away) and Histogram (with virtually any number and range of data cells).

Perhaps now you can begin to see how all of these capabilities make it easy to program with SCAT:

To operate a stimulus device: S1;
To reset a Clock: CLK1<0;
To store its value, at any time, in a Counter: CTR1<CLK1;
To call for a Histogram to be outputted: TYPE HIST1;
To detect a fixed number of responses, say 10, on a given input line: R(10, 1);
To sense when a stored number — perhaps indicating number of trials — equals something, say 50: CTR1.E.50;
And perhaps to do something as a consequence, like terminate the experiment: STOP.

And all this is just the beginning. But please don't let us discourage you from trying to beat the system. Bring your experiment requirements to the Eastern Psychological Association Convention. We'll have SCAT up and running, and ready to take you on. Like we said, if we can't make SCAT fulfill your requirements, to your satisfaction, we'll provide you with a year's subscription to this journal. Just write Dept. 1,
Grason-Stadler Company
West Concord, Mass. 01781 • 617/369-3787.



Try it.