Seconds Before AE	Vehicle Speed (MPH)	Engine Speed (RPM)	Percent Throttle
-5	25	896	0
-4	22	640	0
-3	17	704	0
-2	15	768	0
-1	14	1152	0

Seconds Before AE	Brake Switch Circuit Status
-8	OFF
-7	OFF
-6	OFF
-5	ON
-4	ON
-3	ON
-2	OFF
-1	OFF

Those of you who are familiar with the early 2000-plus GM downloads probably remember the colorful chart which plotted four parameters of vehicle operation for five time increments (typically one second preceding Algorithm Enable--AE). Because these data points were acquired and recorded asynchronously (i.e., not at exactly the same time--variation was plus or minus 0.2 second), the continuous lines which connected each point were misleading, and variation of any value could occur between the recorded data points which would not appear on the chart, but the chart would give the impression that there was a uniform transition or steady state from one data point to the next. The chart is gone. The tables above are from a download from a late-model GM Airbag Control Module (ACM). You probably noticed that brake switch status is now listed for 8 seconds before AE. The reason has to do with binary code.

Computers only know two values: zero and one, or on and off. Each unit of data is called one bit. Eight bits together are called a byte. There are other names given to larger collections of data bits. But the brake status is stored in one byte, so we have 8 values of on or off.

The Crash Data Retrieval (CDR) printouts include a table of hexadecimal values. This is a shorthand for representing binary code. The hexadecimal system is actually a base-16 number system (two bytes) in which the numbers 0 through 9 represent values with which we are all familiar. The letter A represents 10; B represents 11; C represents 12; D represents 13; E is for the value 14, and F represents 15. In hexadecimal code, 10 is the same value as the number 16 in the base-10 system with which we are all familiar. The highest number which can be represented by hexadecimal code is FF, which corresponds to a base-10 value of 255. A CDR printout includes all of the hexadecimal codes stored in the memory chip of the ACM at the time of the download. Some of the values may be default values, and some of the values may intentionally not be interpreted by the software; manufacturers do not necessarily provide all of the software coding necessary to translate every byte of data stored on the memory chip. We get what's allowed by the vehicle and module manufacturers.

The chart below is a typical plot of delta-v as a function of time from a late-model GM ACM download. As you can see, this crash pulse was relatively short; it was essentially over 90 milliseconds (0.090 second) after AE. And the total delta-v after AE was less than five mph. Not a bad bump! As those of you who have read my previous newsletters know, delta-v is the sudden speed change which occurs during the contact phase of a collision.

Version 3.1 of the Bosch CDR software is now out. With this latest release, four new cables were added. You may browse the list of supported vehicles and modules on my Web site using **www.ralphcunningham.net/ver3list.pdf**. The character after the 3 in that link is the letter "l," not the number "1." (They appear identical in so many fonts!)

In addition to adding new vehicles and modules in the upgrade from version 3.0 to 3.1 was a correction to a problem which caused erroneous data to fill the engine speed and road speed rows of data downloads from certain modules from Chrysler Corporation vehicles. Data flows through conductors in streams; markers are placed in those streams to identify start points, stop points, and other critical elements in the data stream. It seems that the manufacturer of the modules in the Chrysler vehicles which had the erroneous data had provided incorrect information concerning the placement of the markers for those data streams; all the bits were in place,