Alarm Scanning Info

Active list statistics

Tue, Jul 15, 2008

Each front end keeps a list of active channels and bits for increased efficiency in performing alarm scanning, which is done at 15 Hz. This is especially important for PowerPC front ends, for which access to nonvolatile memory tables is very slow, about 1 μs per access. A separate note, *Alarm Scanning for PPC*, describes this scheme, which was implemented for both 68K and PowerPC front ends. This note describes a page application `ALRM` that can list out some related statistics for a node or family of nodes, thus making it easy to monitor the extent of alarm scan activity in each front end.

Index arrays are allocated for holding arrays of the active channels and active bits, *i.e.*, those that are in the alarm scan, implying for each one that its Active alarm flag bit is set. (In Acnet, this implies that the signal is "not bypassed.") There is a separate allocated array for channels and for bits. (There is also one for comments, although it always specifies two comment alarms to check, which are all we have ever used.)

The allocated block contains a 32-byte header preceding the array , as follows:

Field	*Size*	*Meaning*
`MBlkSize`	2	Size of allocated block
`MBlkSpar`	4	(spare words)
`MBlkType`	2	Memory block type# =`000A`
`ActivT`	2	Time to perform reconstruction of active entry array in μs
`ActivN`	2	Current #active entries
`ActivMn`	2	`ActivN` minimum
`ActivMx`	2	`ActivN` maximum
`ActivC`	4	Count of alarm-flag-change-initiated reconstructions
`ActivCB`	4	Count of periodic backup reconstructions
`ActivDT`	8	BCD Date/time of last non-backup reconstruction

The tricky part here is finding where these allocated blocks reside. It can be done manually using the facilities of the Memory Dump page, but it's a bit tedious, especially when one has to do it for more than a single node. The method is described in the above-mentioned note. Page application `ALRM` does it the same tedious way, but it does it via software, so it is much faster—not to mention easier on the user! The method starts with the `Alarms` task entry in the Task Table. From the Task ID, or `TID`, it looks up the `Alarms` task local variables, finding pointers to the allocated blocks of interest. The method is only slightly different for 68K nodes as for PowerPC nodes. Upon collecting the allocated block headers, it extracts from them a few relevant items of interest, listing them on a single line via the serial port, targeting the serial port of a target listing node. These are the pieces of data listed for each node.

Field	*Meaning*
`NODE`	Front end node#
`NCHAN`	#active analog channels
`NBIT`	#active bits
`BTIM`	Time to rebuild the allocated array of active channel#s, in ms
`BDAT`	Time-of-day when the array of channels was last rebuilt

Here is an example of the alarm statistics listed for all Linac nodes:

```
4 ALARM STATS     07/15/08 1418
NODE<NLIN>            LIST<0509>
NODE CHAN BIT BTIM LAST-BUILD
0600   2   1  1.29 10/03/07 0700
0601   4   1  0.14 10/03/07 0700
060B   1   0  0.57 07/10/08 2045
0610  73  81  0.3  07/14/08 1059
0611  90  58  0.1  07/14/08 1059
0612  66  81  0.4  06/10/08 1633
0613  62  81  0.44 07/14/08 1059
0614  51  52  0.5  07/10/08 1443
0615  48  81  0.2  07/11/08 1540
0617   3  53  0.89 10/03/07 0700
061C  15  19  0.44 03/30/08 1251
061E  27  29  1.62 06/17/08 1104
0620   2  98  0.67 05/22/08 1408
0621   2  46  0.72 10/03/07 0700
0622   2  48  0.89 05/22/08 1408
0623   2  50  1.48 05/22/08 1408
0624   2  73  0.67 05/22/08 1408
0625   2  65  1.1  05/22/08 1408
0626   2  70  1.07 06/03/08 2138
0627   3  67  0.93 05/22/08 1408
062D   1  44  0.83 10/03/07 0700
0628   3   0  1.02 04/18/08 0923
062C  12   1  1.97 07/11/08 1248
062E 108   4  0.76 07/08/08 1342
062F  47   1  0.48 07/15/08 0934
0619   3   0  0.1  10/03/07 0700
0721   3   2  0.8  05/22/08 1408
0722   3   2  1.4  05/22/08 1408
0723   3   2  0.88 05/22/08 1408
0724   3   2  1.14 05/22/08 1408
0725   3   2  0.89 05/22/08 1408
0726   3   2  1.44 06/04/08 0150
0727   4   2  1.65 05/22/08 1408
0728   1   0  1.15 07/10/08 1143
0730 148  98  0.43 07/11/08 1445
0731  93  47  0.95 05/22/08 1428
0732  97  48  1.23 05/22/08 1408
0733  92  50  1.32 05/22/08 1408
0734  97  73  0.51 05/31/08 0649
0735  93  65  0.93 06/04/08 1640
0736 108  71  0.43 06/03/08 2130
0737 118  67  0.49 05/22/08 1408
073D  90  44  0.9  07/11/08 1640
0602  15   1  0.59 05/22/08 1408
0616   2   1  0.58 03/30/08 1250
061F  10   1  0.61 06/10/08 1958
```

One thing noticeable is that many nodes do not have frequent changes made to the list of active channels. For example, `node061E` has not had such a change for more than a month.

The time required to rebuild the allocated array of channels varies a bit. It should depend primarily on the number of allocated channels, which is 1024 in some nodes and 2048 in others. It may be that in some cases, the time is affected by interrupt activity.

The alarm scanning time, which is of obvious interest, is not shown here. It can be found by

using a different page application, PAGETASK, taking care to run it in a node of the same type. (The 68K and PowerPC versions of this PA are not quite compatible.) One example of alarm scanning time for a PowerPC node is 0.21 ms, for node0730, which now has 148 channels and 98 bits to scan. An example for a 68K node is 0.47 ms, for node058A, which has 128 channels and 0 bits to scan. These two nodes are among those with the largest active channels, bits.

Screen layout

Here is an example of the little console display after scanning the ECool nodes:

```
4 ALARM STATS     07/15/08 1420
NODE<NECL>  7         LIST<0509>
NODE CHAN BIT BTIM LAST-BUILD
0582  16   0  0.61 07/11/08 1553
058A 128   0  0.58 07/03/08 1250
058C 123   0  0.66 06/10/08 1519
058D   5   0  0.61 04/12/08 1426
058E   7   0  0.63 04/12/08 1426
05E8   5   0  0.7  04/14/08 1747
05E9  33   4  0.78 06/11/08 1311
```

Up to 12 nodes can be shown on the display. When there are more than 12, the display simply "wraps." The count of nodes scanned is given just to the right of the NODE field.