GMT Support Test

Robert Goodwin
Mon, Feb 8, 2010

Calendar time

In 2002, support was added to the front end systems to support calendar time, referred to as GMT time, consisting of two 32-bit words, the first giving the unsigned number of seconds since Jan 1, 1900, and the second giving microseconds into that calendar second. The GMT time for the 15 Hz `0x0C` clock event is developed and shared via multicast to all front ends once per minute. See the note, *Calendar Time*, for more on this support.

New diagnostic

A new local application called `EV8F` has been written to monitor the GMT times against the occurrences of clock event `0x8F`, which is designed to occur on each calendar second. A function called `EventGMT` returns the GMT time of any given clock event, based upon the local front end's knowledge of the GMT time of the most recent `0x0C` event. It uses the local Event Times table to derive this GMT time for any given event.

Parameter layout

The parameters for `EV8F` are as follows:

Field		*Size*	*Meaning*
`ENABLE`	`B`	2	Usual Enable Bit#
`DEVCHAN`	`C`	2	Deviation of event `0x8F` from the one second mark Chan#, in μs
`THRESH`		2	Threshold for internal logging, in μs

Program logic flow

Call `HaveEvt` to detect the occurrence of the `0x8F` event. When this function returns `true`, it means that event `0x8F` has occurred within 66 ms before the start of the current front end's operating cycle. For example, MiniBooNE front ends begin their operating cycle at 40 ms after Booster reset event time, called Micro-P Start. (This ensures that they start after Booster beam extraction, which occurs about 35 ms after Booster reset event time.) When the clock event `0x8F` is detected, `EV8F` calls `EventGMT` to get the calendar time for that event. It then looks at the μs component of that time structure to see how near it is to an exact second mark, meaning that it will be either very near to zero, or very near to 999999 μs. It works out the deviation from the second mark, which is the value to be examined. Any change that exceeds the threshold parameter is logged. When this was first tried, the threshold parameter was set for 100 μs.

Rather than log every deviation exceeding the threshold, the logic actually watches for the change in the deviations to exceed the threshold. In this way, if a long sequence of `0x8F` events yielded returns from `EventGMT` for which the μs component strays from the second mark by, say, 200 ms, only one entry would be logged when this started happening, and another if and when it ceased happening. This kind of "change" logic is done recognizing that the NTP server is queried by the `TIME` local application only once per minute, when it computes the GMT time for the `0x0C` event that is shared with all front ends. If, for whatever reason, a single bad reply came from the NTP server, this would otherwise result in 60 entries being logged by `EV8F`.