

# Event Comings and Goings

*Local application*

Robert Goodwin

Tue, Dec 15, 2009

Local application `ENEW` was developed as another clock event diagnostic, about which this note is written. It monitors clock events placed in the `EVTLOG` data stream of a target node, watching for changes in the pattern of clock events. When a new event occurs that has not been seen for awhile, it logs its occurrence. Likewise, when it sees that an active event has *not* been seen in awhile, it logs that cessation. The meaning of “awhile” is herein referred to as a period, which is specified as a parameter to this LA. For initial testing, this parameter was set for a 10 minute period.

The internal log is part of the “static memory block” allocated by the LA upon initialization. A log entry includes the clock event#, the time, and a count of events seen. The count is zero for a “new” event, and it is nonzero for a cessation entry. Here is the 8-byte log entry format:

```
evDa HrMn ScCy cntr
```

The first byte is the clock event#, the next 5 bytes give the BCD time, and the `cntr` word is the count, limited to `0xFFFF`, or 65535. Here, via Print Memory PA, is an example of part of such a log:

```
:000738BC    E014 1325 0307 0000  *
:000738C4    9A14 1325 0307 0000
:000738CC    9414 1325 0501 0000
:000738D4    F014 1325 0801 0000
:000738DC    EE14 1325 0807 0000
:000738E4    9314 1332 4601 0004
:000738EC    1614 1332 4602 0004
:000738F4    9914 1332 4710 0004
:000738FC    9514 1332 4910 0004
:00073904    9114 1338 1200 0002
:0007390C    2D14 1338 2200 0006
:00073914    ED14 1338 2200 0006
:0007391C    7D14 1338 2201 0006
:00073924    9A14 1338 2201 0002
:0007392C    E014 1338 2201 0002  *
:00073934    9414 1338 2310 0002
:0007393C    F014 1338 2610 0002
:00073944    EE14 1338 2701 0002
:0007394C    EC14 1341 0905 0005
:00073954    DB14 1407 0408 0008
:0007395C    DB14 1407 0409 0000
```

The times show that all of these entries were logged today, in less than an hour. Take the first entry specifically. It indicates that a first occurrence of event `0xE0` was at 1325:03. Later, at 1338:22, none had been seen for 10 minutes; thus, the last of two `0xE0` events occurred at 1328:22. Similar statements can be made about events `0x9A`, `0x94`, `0xF0`, and `0xEE`. At the end, note an entry for event `0xDB` that indicates that at 1407:04, 10 minutes had passed since the last such event occurred. But on the very next cycle, another such event was seen. Apparently, `0xDB` events may occur with 10 minute spacing.

An alternate view of the log entries can be done by sorting the entries by 8-byte content. To do

this, use the following regular expression to remove the (unneeded) addresses from the list, then sort the list in the usual way.

```
: [0-9A-F ]+
```

This places all log entries about each event (in event# order) in chronological sequence, thus giving a picture of the comings and goings of each event. Note that many events will not appear in such a log, especially those that occur with regular frequency more often than once/period. Here is a small part of such a sorted list, showing the comings and goings of event NTF beam event 0x09.

```
0915 0828 5509 0000
0915 0838 5906 0019
0915 0857 5009 0000
0915 0959 1008 4BEC
0915 1018 0214 0000
0915 1049 0201 2465
```

Evidently NTF was operating today from about 0828 for 25 pulses over 3 seconds. Half an hour later it ran for about an hour. Later it ran from 1018 until 1039.

### *Internal details*

The parameter layout for ENEW is as follows:

<i>Parameter</i>	<i>Value</i>	<i>Meaning</i>
ENABLE	B <00AA>	Usual enable Bit#
TARGNODE	<0600>	Target node#
PERIOD	<0258>	Period, in seconds

Note that one can run this program on a test node, targeting an operational node. A 15 Hz data request is made for the most recent entries of the EVTLOG data stream, a convenient way to review all clock events seen by the target node. For each of the 256 possible events, a count and a cycle number of its last occurrence is maintained. When an event is seen that has a zero count, a new entry is logged. All 256 events are reviewed each cycle for detecting when an entire period has transpired since the last event occurrence, so that a cessation entry is logged.

Running in a 68K front end, this LA executes in about 0.3 ms on each 15 Hz cycle. The internal log, holding space for a 256-entry circular buffer is located in the static memory block at offset 0x100.