## Interpreting Real-Time-Clock Registers


#### Abstract

This application note describes the format used for Maxim Real-Time Clock (RTC) time and date registers.


Maxim RTCs (real-time clocks) provide both current clock and calendar information to the end user. Many of these RTCs have a time/date alarm function that can be programmed for periodic timed alarms or for a specific one-time calendar alarm. This application note is intended to ease the implementation of Maxim RTCs through detailed discussions and examples of how to properly configure and use the clock, calendar, and alarm functions. Discussions of Timekeeping or Alarm Register contents will reference decimal equivalents of time/date data contained within, unless otherwise specified.

## RTC Clock and Calendar Functions

Maxim RTCs provide both clock and calendar functions. Illogical settings will not produce predictable results. The RTC algorithms for clock and calendar counter functions are based on valid data being entered into the respective registers. The old engineering saying of "garbage in, garbage out" holds true here. From both cost and size perspectives, there are no extensive checks and balances for the data entered into any of the Timekeeping Registers used to generate clock and calendar data. It is left to the user's discretion to enter only valid clock and calendar information into the RTC Timekeeping Registers.

## Table 1. Typical RTC Timekeeping Registers



| HR | 1 | 0 | 0 | 0 | 0 | 1 |  |  | $\begin{aligned} & 01-12 \\ & 00-23 \end{aligned}$ |  | 12/24 |  | $\begin{aligned} & 10 \mathrm{R} \\ & \mathrm{~A} / \mathrm{P} \end{aligned}$ | 10HR | 1HR |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  | 0 |  |  |  |  | 1/0 | 0 |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  | *POR STATE | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DATE | 1 | 0 | 0 | 0 | 0 |  | 1 | 1 |  | 01- <br> 28/29 <br> 01-30 <br> 01-31 <br> *POR <br> STATE | 0 | 0 | 10 DATE |  | 1 DATE |  |  | 1 |
|  |  |  |  |  |  |  | 0 |  |  |  | 0 | 0 | 0 | 0 | 0 | 0 |  |
|  | 1 | 0 | 0 | 0 | 1 | 0 |  | 0 | $\begin{aligned} & \text { RD } \\ & \hline \text { W } \end{aligned}$ |  | 01-12 | 0 | 0 | 0 | 10 M | 1 MONTH |  |  |  |
| MONTH |  |  |  |  |  |  | *POR STATE |  |  |  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
|  | 1 | 0 | 0 | 0 | 1 | 0 | 1 |  | $\begin{aligned} & \mathrm{RD} \\ & \mathrm{~W} \end{aligned}$ | 01-07 | 0 | 0 | 0 | 0 | 0 | WEEKDAY |  |  |
| DAY |  |  |  |  |  |  |  |  | $\begin{aligned} & \text { *POR } \\ & \text { STATE } \end{aligned}$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |  |
|  | 1 | 0 | 0 | 0 | 1 |  | 1 | 0 |  |  | 00-99 | 10 YEAR |  |  |  | 1 YEAR |  |  |  |
| YEAR |  |  |  |  |  |  | W |  | *POR STATE | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |  |
|  | 1 | 0 | 0 | 0 | 1 | 1 |  | 1 | 1 R | RD |  | WP | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| CONTROL |  |  |  |  |  |  |  |  | W | *POR STATE | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |


|  | 1 | 0 | 0 | 1 | 0 | 0 | 0 | RD |  | TCS | TCS | TCS | TCS | DS |  | RS |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| CHARGER |  |  |  |  |  |  |  |  | state |  | 0 | 0 | 0 | 0 | 0 | 0 |  |  |


| CENTURY | 1 | 0 | 0 | 1 | 0 | 0 | RD |  |  | 1000 YEAR |  |  |  |  | 100 YEAR |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  |  |  |  | $\begin{aligned} & \text { *POR } \\ & \text { STAT } \end{aligned}$ |  | 0 | 0 |  |  | 1 | 0 | 0 | 1 |

[^0]| CONFIG |  |  |  |  |  |  |  |  | /W | *POR STATE | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 1 | 0 | 0 |  | 1 | 0 | 1 | 1 | RD |  | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| CONFIG |  |  |  |  |  |  |  |  | /W | *POR STATE | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

Note: *POR STATE defines the Power-On Reset state of register content.

## Seconds Register

The Timekeeping Seconds Register contains the current time count from the seconds counter. The seconds-counter output ranges from 00 to 59 seconds and carries over to 00 on the next 1 -second count following a previous count of 59. Seconds data is contained in D6 through D0, and BCD format applies to those data bits. D7 is used in some RTCs for a status or control bit. In those RTCs, reads to and writes from the Seconds Register must mask D7 and interpret only D6 through D0 as seconds data, as shown below.

## Table 2. Timekeeping Seconds Register

| Seconds <br> Register | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  | $\mathbf{0}$ | 10SEC (BCD 0-5) <br> (Decimal | 0-5) | 1SEC (BCD 0-9) <br> (Decimal |  |  |  |  |
| 0-9) |  |  |  |  |  |  |  |  |

Note: Valid seconds data $=00$ to 59

## Minutes Register

The Timekeeping Minutes Register contains the current time count from the minutes counter. The minutes-counter output ranges from 00 to 59 minutes and carries over to 00 on the next 1-minute count following a previous count of 59. Minutes data is contained in D6 through D0, and BCD format applies to those data bits. D7 is used in some RTCs for a status or control bit. In those RTCs, reads to and writes from the Minutes Register must mask D7 and interpret only D6 through D0 as minutes data, as shown below.

Table 3. Timekeeping Minutes Register

| Minutes <br> Register | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  | $\mathbf{0}$ | 10MIN (BCD <br> (Decimal 0-5) | 1MIN (BCD <br> (0-9) <br> (Decimal |  |  |  |  |  |
| Min Binary Count | $\mathbf{0}$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Max Binary <br> Count | $\mathbf{0}$ | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| POR STATE | $\mathbf{0}$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Note: Valid minutes data $=00$ to 59

## Hours Register

The Timekeeping Hours Register contains the current time count from the hours counter. There are two operating modes for the Hours Register: 12-hour format and 24-hour format. The interpretation of D5 depends on the hour format selected in D7.

Table 4. Timekeeping Hours Register

| Hours Register | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |  |  |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :---: | :---: |
|  | $12 / 24$ | 0 | 10HR | 10HR | 1HR |  |  |  |  |  |
|  | $1 / 0$ |  | A/P $0 / 1$ |  |  |  |  |  |  |  |
|  | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |  |

D7, the 12/24 format bit set to a 0 , configures the hours counter for 24 -hour format. D5 is used as the upper bit in the 10 -hour BCD format. Hours data in the 24 -hour format is contained in D5 through D0, and BCD format applies to those data bits. In this mode, the hours-counter output ranges from 00 to 23 hours and carries over to 00 on the next 1-hour count following a previous count of 23.

Table 5. Timekeeping Hours Resister: 24-Hour Format

| Hours <br> Register <br> (24HR | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
| Format) |  |  |  |  |  |  |  |  |


| Max Binary <br> Count | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |

Note: Valid hour data (24-hour format) $=00$ to 23
D7, the 12/24 format bit set to a 1 , configures the hours counter for 12-hour format. D5 is used as the $A M / P M$ indicator with $1=P M$ and $0=A M$. Hours data in the 12 -hour $A M$ format or in the 12hour PM format is contained in D4 through D0, and BCD format applies to those data bits. In this mode, the hours-counter output ranges from 01 to 12 hours and carries over to 01 on the next 1hour count following a previous count of 12. In addition, the AM/PM bit changes on the hour increment from 11 PM $(\mathrm{D} 5=0)$ to $12 \mathrm{AM}(\mathrm{D} 5=1)$ and from $11 \mathrm{AM}(\mathrm{D} 5=1)$ to 12 PM ( $\mathrm{D} 5=0$ ).

Table 6. Timekeeping Hours Register: 12-Hour AM Format

| Hours Register (12HR AM Format) | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 1 (12HR) | 0 | 0 (AM) | $\begin{aligned} & \text { 10HR (BCD } \\ & 0-1 \text { ) } \\ & \text { (Decimal 0- } \\ & \text { 1) } \end{aligned}$ |  | $\begin{aligned} & \text { (BCD } \\ & \text { cimal } \end{aligned}$ | $\begin{aligned} & \text { 1-2) } \\ & 1-2) \end{aligned}$ |  |
| Min Binary Count | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Max Binary Count | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |

Note: Valid hour data (12-hour format) = 01 to 12
Table 7. Timekeeping Hours Register: 12-Hour PM Format

| Hours Register (12HR PM Format) | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 1 (12HR) | 0 | 1 (PM) | $\begin{aligned} & \text { 10HR (BCD } \\ & 0-1) \\ & \text { (Decimal 0- } \\ & \text { 1) } \end{aligned}$ |  | imal | $\begin{aligned} & \text { D 1-2) } \\ & (1-2) \end{aligned}$ |  |
| Min Binary Count | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |


| Max Binary <br> Count | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |

Note: Valid hour data (12-hour format) $=01$ to 12

## Date Register

The Timekeeping Date Register contains the current calendar day of the month from the date counter. The counter-output range is dependent on the current value of the Month Register, Year Register, and Century Register. The date-counter range is 01 to 28 , 01 to 29, 01 to 30, or 01 to 31. At maximum count, the date counter carries back to 01 on the next day increment computed from the Hours Register. Date data is contained in D5 through D0, and BCD format applies to those data bits. If D6 and D7 are always 0 , which is the case for most Maxim RTCs, then BCD format applies for the entire Date Register.

## Table 8. Timekeeping Date Register

| Date Register | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 0 | 0 | 10 DATE (BCD 0-1 DATE (BCD 1-9) <br> 3) (Decimal 0-3) (Decimal 1-9) |  |  |  |  |  |
| Min Binary Count | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Max Binary Count | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| POR STATE | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Note: Valid date data = 01 to 31,01 to 30,01 to 29, 01 to 28
Table 9. Counting Range of Date Counter

| Range | Carry | Month Register (BCD) | Year and Century Register |
| :---: | :---: | :---: | :---: |
| 01 to 31 |  | 01 (January) 03 (March) 05 (May) 07 (July) 08 (August) 10 (October) 12 (December) | N/A |


| 01 to 30 | 04 (April) <br> 06 (June) <br> 09 (September) <br> 11 (November) | N/A |
| :--- | :--- | :--- |
| 01 to 29 | 02 (February) | Divide by 4 with <br> no remainder |
| 01 to 28 | 02 (February) | Divide by 4 with <br> remainder |

## Month Register

The Timekeeping Month Register contains the current time count from the month counter. The month-counter output ranges from 01 to 12 months and carries over to 01 on the next 1-month count following a previous count of 12 . Month data is contained in D4 through D0, and BCD format applies to those data bits. If D7, D6, and D5 are always 0 , which is the case for most Maxim RTCs, then BCD format applies for the entire Date Register.

Table 10. Timekeeping Month Register

| Month <br> Register | D7 | D6 D5 | D4 | D3 | D2 | D1 | D0 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ |  |  |  |  |

Note: Valid month data = 01 to 12
Table 11. Month Register (BCD)

| Month <br> Register <br> (BCD) | Month |
| :--- | :--- |
| 01 | January |
| 02 | February |
| 03 | March |


| 04 | April |
| :--- | :--- |
| 05 | May |
| 06 | June |
| 07 | July |
| 08 | August |
| 09 | September |
| 10 | October |
| 11 | November |
| 12 | December |

## Day Register

The Timekeeping Day Register (day of the week) contains the current weekday count from the day counter. The day-counter output ranges from 01 to 07 days and carries over to 01 on the next 1-day count following a previous count of 07 . Weekday data is contained in D4 through D0, and BCD format applies to those data bits. If D7 through D3 are always 0 , which is the case for most Maxim RTCs, then BCD format applies for the entire Day Register.

Table 12. Timekeeping Day Register

| Day Register | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | WEEKDAY (BCD 1- <br> 7) (Decimal 1-7) |  |  |
| Min Binary <br> Count | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | 0 | 0 | 1 |
| Max Binary <br> Count | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | 1 | 1 | 1 |
| POR STATE | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | $\mathbf{0}$ | 0 | 0 | 1 |

Note: Valid day data $=01$ to 07

## Year Register

The Timekeeping Year Register contains the current units and 10's count for the year counter. The year-counter output ranges from 00 to 99 and carries over to 00 on the next 1 -year count following a previous count of 99 . The current year units' count is contained in D3 through D0 in BCD format. The current year 10's count is contained in D7 through D4 in BCD format.

Table 13. Timekeeping Year Register

| Year <br> Register | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  | 10 YEAR (BCD <br> (Decimal) | 1 YEAR (BCD <br> (Decimal 0-9) |  |  |  |  |  |  |
| Min Binary <br> Count | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Max Binary <br> Count | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| POR STATE | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |

Note: Valid year data $=00$ to 99

## Century Register

The Timekeeping Century Register contains the current thousand's and hundred's count for the century counter. The century-counter output ranges from 00 to 99 and carries over to 00 on the next 1 -century count following a previous count of 99 . The current year thousand's count is contained in D7 through D4 in BCD format. The current year hundred's count is contained in D3 through D0 in BCD format.

Table 14. Timekeeping Century Register

| Century Register | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
|  |  |  |  |  |  |  |  |  |
|  | 1000 YEAR |  |  | 100 YEAR |  |  |  |  |
| Min Binary Count | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Max Binary Count | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| POR STATE | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

Note: Valid century data $=00$ to 99

## Time/Date Example

Table 15. Register Contents for March 26, 2001; Monday; 11:24:36 PM (23:24:36 hours)

| Hex <br> Data | BCD <br> Data | Decimal <br> Timekeeping <br> Contents | Register/ <br> Binary <br> Data | D7 | D6 | D5 | D4 | D3 | D2 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | D1 | D0 |
| :--- |


| Seconds | Seconds | Seconds | Seconds | 0 | 10SEC |  |  | 1SEC |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 36h | 36 | 36 | Binary | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Minutes | Minutes | Minutes | Minutes | 0 | 10MIN |  |  | 1 MIN |  |  |  |
| 24 h | 24 | 24 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| Hour (24HR Format) | Hour (24HR Format) | Hour (24HR Format) | Hour (24HR Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | $\begin{aligned} & 10 \\ & \text { HR } \end{aligned}$ | $\begin{aligned} & 10 \\ & \text { HR } \end{aligned}$ | 1 HR |  |  |  |
| A3h | A3 | 23 | Binary | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| Hour <br> (12HR <br> Format) | Hour <br> (12HR <br> Format) | Hour (12HR Format) | Hour (12HR Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | $\begin{aligned} & \mathrm{A} / \mathrm{P} \\ & (0 / 1) \end{aligned}$ | $\begin{aligned} & 10 \\ & \mathrm{HR} \end{aligned}$ | 1 HR |  |  |  |
| 31h | 31 | 11 PM | Binary | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| Date | Date | Date | Date | 0 | 0 | $\begin{aligned} & 10 \\ & \text { DATT } \end{aligned}$ |  | 1 DATE |  |  |  |
| 26h | 26 | 26 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| Month | Month | Month | Month | 0 | 0 | 0 | $\begin{aligned} & 10 \\ & \mathrm{M} \end{aligned}$ | 1 MONTH |  |  |  |
| 03h | 03 | 3 (March) | Binary | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Day | Day | Day | Day | 0 | 0 | 0 | 0 | 0 WEEKDAY |  |  |  |
| 02h | 02 | 2 (Monday) | Binary | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Year | Year | Year | Year | 10 YEAR |  |  |  | 1 YEAR |  |  |  |
| 01h | 01 | 01 | Binary | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| Century | Century | Century | Century | 1000 YEAR |  |  |  | 100 YEAR |  |  |  |
| 20 h | 20 | 20 | Binary | 0 | 0 | 1 | 0 |  | 0 | 0 | 0 |

## RTC Alarm Function

For Maxim RTCs that provide an alarm function, there are a variety of possible configurations for the alarm settings. lllogical settings will not produce predictable results. From both cost and size
perspectives, there are no extensive checks and balances for the data entered into the Alarm Configuration or Alarm Threshold Registers. It is left to the user's discretion to enter only valid clock and calendar information into the RTC Alarm Registers.

## RTC Alarm Configuration Register

The RTC Alarm Configuration Register contains bits for programming when a date/time alarm occurs.

## Table 16. Alarm Configuration Register

| Register Acd | dr | ess | C | nr |  |  |  |  |  | Registe | r |  | finiti |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| FUNCTION |  | A6 | A | A | A3 | A |  | 1 | A0 | VALUE | D | 7 D |  | D5 | D4 | D3 | D2 | D1 | D0 |
| CLOCK |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 1 | 0 | 0 | 1 | 0 | 1 | 0 |  | RD |  | 0 |  | YEAR | DAY | MONTH | DATE | HR | MIN | SEC |
| CONFIG |  |  |  |  |  |  |  |  | /W | $\begin{aligned} & \text { *POR } \\ & \text { STATE } \end{aligned}$ | 0 | 0 |  | 0 | 0 | 0 | 0 | 0 | 0 |

Note: *POR STATE defines the Power-On Reset state of register content.

A logic 0 in any bit does not enable a comparison of the respective Alarm Threshold Register with the associated timekeeping-counter output for that respective time unit. A logic 1 enables the comparison. Each bit set to a logic 1 in the Alarm Configuration Register must have conditions satisfied for the alarm to occur. For example, if D0, seconds bit, is set to 1 and D1, minutes bit, is set to 1 , then the alarm will occur every time the timekeeping counter matches both the count stored in the Seconds Threshold Register and the count stored in the Minutes Threshold Register.

When the Alarm Threshold Registers, which have been selected by the Alarm Configuration Register, match their respective timekeeping-counter outputs, the alarm will trigger with a typical 3 mS delay. Once the alarm is triggered, it can be cleared by writing or reading to any Alarm Threshold Register or by reading or writing to the Alarm Configuration Register.

Because the alarm-trigger function is an edge-triggered logic, the alarm only occurs once each time the respective counter reaches the count that matches the respective Alarm Threshold Register setting. For example, set the Alarm Configuration Register to only minutes by setting D1 to logic 1 and all other data bits to zero. Now, in the Alarm Threshold Minutes Register, set its LSB to 1 and all other bits to zero. This configures the RTC alarm logic to trigger every time the minutes counter transitions from 00 to 01 . The first transition of the minutes counter from 00 to 01 sets the alarm. If the alarm is now cleared by reading or writing to the Alarm Configuration Register or any of the Alarm Threshold Registers, it will not be reset again until the next edge transition of the minutes counter from 00 to 01 , regardless of whether or not the output of the minutes counter is still at 01.

Typical RTC Alarm Threshold Registers are shown below. The Alarm Threshold Registers (Year,

Day, Month, Date, Hours, Minutes, Seconds) have the same data configuration as their respective Timekeeping Register counterparts, described in detail earlier.

Table 17. Typical RTC Alarm Registers

## Register/Address Command

 Register DefinitionFUNCTION A7 A6 A5 A4 A3 A2 A1 A0 VALUE D7 D6 D5 D4 D3 D2 D1 Do ALARM
THRESHOLDS



DAY

|  |  |  |  |  |  |  |  |  | /W | *POR STATE | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 1 | 0 | 1 | 0 | 0 |  | 1 | 0 | RD | 00-99 |  | EAR |  |  |  | YEA |  |  |
| YEAR |  |  |  |  |  |  |  |  | /W | *POR STATE | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| CLOCK BU/RST | 1 | 0 | 1 | 1 | 1 |  | 1 | 1 | $\begin{aligned} & \mathrm{RD} \\ & \hline \mathrm{~W} \end{aligned}$ |  |  |  |  |  |  |  |  |  |
| RAM (VBATT Backed Up) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| RAM 0 | 1 | 1 | 0 | 0 | 0 |  | 0 | 0 | RD | RAM DATA 0 | x | X | x | X | X | X | x | x |
|  |  |  |  |  |  |  |  |  | /W |  |  |  |  |  |  |  |  |  |
| ] |  |  | [] |  |  |  |  |  |  | I] |  |  |  | [] |  |  |  |  |
| RAM 30 | 1 | 1 | 1 | 1 | 1 |  | 1 | 0 | RD | RAM DATA 30 | x | X | X | X | X | X | x | x |
|  |  |  |  |  |  |  |  |  | /W |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| RAM BU/RST | 1 | 1 | 1 | 1 | 1 |  | 1 | 1 | $\begin{aligned} & \mathrm{RD} \\ & \mathrm{~W} \end{aligned}$ |  |  |  |  |  |  |  |  |  |

Note: *POR STATE defines the Power-On Reset state of register contents.

## Alarm Examples

There are 128 possible alarm settings using the Alarm Configuration Register. Not all of these possible combinations are logical or cause the desired effect for triggering the alarm (for example, setting an alarm to trigger every April 31). A few common examples are presented in detail below as guidance in correctly configuring the Alarm Configuration Register and the Alarm Threshold Registers.

## Table 18. Summary of Common Examples

## Alarm Configuration Register

| Example | D7 | D6 Year | $\begin{array}{\|l\|} \hline \text { D5 } \\ \text { Day } \end{array}$ | D4 Month | D3 Date | D2 Hours | D1 Minutes | D0 Seconds | Alarm <br> Occurrence <br> (when <br> threshold- <br> register <br> contents <br> match <br> respective timekeepingcounter output) |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | Once every Seconds match (e.g., once a minute @ 30 seconds count) |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | Once every Minutes match (e.g., once an hour @ 45 minutes count) |
| 3 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | Once every Hours* match (e.g., once a day - every 24 hours @ 11 PM) |
| 4 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | Once every Hours*, <br> Minutes, and Seconds match (e.g., once a day every 24 hours @ specific time of day) |
| 5 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Once every Date match (e.g., once a month) |


| 6 | 0 | 0 |  | 1 | 0 | 0 | 0 | 0 | 0 | 0 | Once every Day match (e.g., once a week - every Monday) |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 7 | 0 |  |  | 0 | 1 | 0 |  | 0 | 0 | 0 | Once every Month match (e.g., once/year every March 1 @ 12 AM) |
| 8 | 0 |  |  | 0 | 1 | 1 |  | 1 | 1 | 1 | Once every Time, Date, and Month (Month, Date, Hours*, Minutes, Seconds) match (e.g., once a year specific month, date, time of day*) |
| 9 | 0 |  |  | 0 | 1 | 1 |  | 1 | 1 |  | Once every Time, Date, Month, and Year match (e.g., one-time alarm set specific month, date, year, time of day*) |

Note: *The Alarm Threshold Register must use the same hour format (12-hour or 24 -hour) as the Timekeeping Register, otherwise the alarm function will not operate properly.

## Alarm Example 1: Triggering the Alarm Once Every Minute @ 30-Seconds Count

Once every Seconds match. Only the Alarm Threshold Seconds Register is set, and all other Alarm Threshold Registers (Year, Day, Month, Date, Hours, Minutes) have their internal bits set to logic 0.

Set the alarm to trigger once every minute at the 30 -seconds count.

Table 19. Register Contents for the Alarm Trigger Every Minute @ 30-Seconds Count

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Seconds | Seconds | Seconds | Seconds | 0 | 10S | EC |  | 1S |  |  |  |
| 30h | 30 | 30 | Binary | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |

Alarm Example 2: Triggering the Alarm Once Every Hour @ 45-Minutes Count
Once every Minutes match. Only the Alarm Threshold Minutes Register is set, and all other Alarm Threshold Registers (Year, Day, Month, Date, Hours, Seconds) have their internal bits set to logic 0.

Set the alarm to trigger once every hour at the 45-minutes count.
Table 20. Register Contents for the Alarm Trigger Every Hour @ 45-Minutes Count

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Minutes | Minutes | Minutes | Minutes | 0 |  | MIN |  | 1M |  |  |  |
| 45h | 45 | 45 | Binary | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |

Alarm Example 3 (24-Hour Format): Triggering the Alarm Once a Day @ 23:00:00 Hours
Once every Hours match. Only the Alarm Threshold Hours Register is set, and all other Alarm Threshold Registers (Year, Day, Month, Date, Minutes, Seconds) have their internal bits set to logic 0 . The Alarm Threshold Hour Register must use the same hour format (12-hour or 24 -hour) as the Timekeeping Hour Register, otherwise the alarm function will not operate properly.

Set the alarm to trigger every 24 hours (once a day) at 23:00:00 hours.

Table 21. Register Contents for the Alarm Trigger Every 24 Hours @ 23:00:00 Hours

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary <br> Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |


| Hour (24HR <br> Format) | Hour (24HR Format) | Hour (24HR Format) | Hour (24HR Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | 10HR | 10HR | 1HR |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| A3h | A3 | 23 | Binary | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |

## Alarm Example 3 (12-Hour Format): Triggering the Alarm Once a Day @ 11 PM

Once every Hours match. Only the Alarm Threshold Hours Register is set, and all other Alarm Threshold Registers (Year, Day, Month, Date, Minutes, Seconds) have their internal bits set to logic 0 . The Alarm Threshold Hour Register must use the same hour format (12-hour or 24-hour) as the Timekeeping Hour Register, otherwise the alarm function will not operate properly.

Set the alarm to trigger every 24 hours (once a day) at 11 PM.
Table 22. Register Contents for the Alarm Trigger Every 24 Hours @ 11 PM

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Hour (12HR Format) | $\begin{aligned} & \text { Hour } \\ & \text { (12HR } \end{aligned}$ Format) | Hour (12HR Format) | Hour <br> (12HR <br> Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | $\begin{aligned} & A / P \\ & (0 / 1) \end{aligned}$ | 10HR | 1HR |  |  |  |
| 31h | 31 | 11 PM | Binary | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |

Alarm Example 4 (24-Hour Format): Triggering the Alarm Once a Day @ 23:24:36 Hours

Once every Hours, Minutes, and Seconds match. Only the Alarm Threshold Hours Register, Minutes Register, and Seconds Register are set, and all other Alarm Threshold Registers (Year, Day, Month, Date) have their internal bits set to logic 0. The Alarm Threshold Hour Register must use the same hour format (12-hour or 24 -hour) as the Timekeeping Hour Register, otherwise the alarm function will not operate properly.

Set the alarm to trigger every 24 hours (once a day) at 23:24:36 hours.

Table 23. Register Contents for the Alarm Trigger Every 24 Hours @ 23:24:36 Hours

| Hex <br> Data | BCD <br> Data | Decimal <br> Timekeeping <br> Contents | Register/ <br> Binary <br> Data | D7 | D6 | D5 | D4 | D3 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |


| Seconds | Seconds | Seconds | Seconds | 0 | 10SEC |  |  | 1SEC |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 36h | 36 | 36 | Binary | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Minutes | Minutes | Minutes | Minutes | 0 | 10MIN |  |  | 1 MIN |  |  |  |
| 24h | 24 | 24 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| Hour (24HR Format) | Hour (24HR Format) | Hour (24HR <br> Format) | Hour (24HR Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | 10HR | 10HR | 1HR |  |  |  |
| A3h | A3 | 23 | Binary | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |

Alarm Example 4 (12-Hour Format): Triggering the Alarm Once a Day @ 11:24:36 PM
Once every Hours, Minutes, and Seconds match. Only the Alarm Threshold Hours Register, Minutes Register, and Seconds Register are set, and all other Alarm Threshold Registers (Year, Day, Month, Date) have their internal bits set to logic 0. The Alarm Threshold Hour Register must use the same hour format (12-hour or 24-hour) as the Timekeeping Hour Register, otherwise the alarm function will not operate properly.

Set the alarm to trigger every 24 hours (once a day) at 23:24:36 hours.
Table 24. Register Contents for the Alarm Trigger Every 24 Hours @ 11:24:36 PM

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register <br> Binary <br> Data | D7 | D6 | D5 | D4 | D3 D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Seconds | Seconds | Seconds | Seconds | 0 |  | SEC |  | 1SEC |  |  |
| 36h | 36 | 36 | Binary | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| Minutes | Minutes | Minutes | Minutes | 0 | 10N | MIN |  | 1 MIN |  |  |
| 24h | 24 | 24 | Binary | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Hour (12HR Format) | Hour (12HR Format) | Hour (12HR <br> Format) | Hour (12HR Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | $\begin{aligned} & \mathrm{A} / \mathrm{P} \\ & (0 / 1) \end{aligned}$ | 10HR | 1HR |  |  |
| A3h | A3 | 23 | Binary | 0 | 0 | 1 | 1 | 00 | 0 | 1 |

Alarm Example 5: Triggering the Alarm Once a Month on the 26th Day

Once every Date match. Only the Alarm Threshold Date Register is set, and all other Alarm Threshold Registers (Year, Day, Month, Hours, Minutes, Seconds) have their internal bits set to logic 0 . Because it is desired for the alarm to trigger each month on the same date, a valid date that exists in each month should be chosen, otherwise the alarm will not trigger during a month that does not contain the particular date (for example, 31 does not exist in June).

Set the alarm to trigger on the 26th day of every month at 12:00:00 AM.
Table 25. Register Contents for the Alarm Trigger Every Month on the 26th Day of the Month @ 12:00:00 AM

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Date | Date | Date | Date | 0 | 0 | $\begin{aligned} & 10 \\ & \text { DAT } \end{aligned}$ |  |  | ATE |  |  |
| 26h | 26 | 26 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |

Alarm Example 6: Triggering the Alarm Once a Week Every Monday @ 12:00:00AM
Once every Day match. Only the Alarm Threshold Day Register is set, and all other Alarm Threshold Registers (Year, Month, Date, Hours, Minutes, Seconds) have their internal bits set to logic 0 .

Set the alarm to trigger once per week on Monday at 12:00:00 AM.
Table 26. Register Contents for the Alarm Trigger Every Week on Monday @ 12:00:00 AM

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Day | Day | Day | Day | 0 | 0 | 0 | 0 | 0 |  | EK | Y |
| 02h | 02 | 2 (Monday) | Binary | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

Alarm Example 7: Triggering the Alarm Once a Year Every March 1 @ 12:00:00 AM
Once every Month match. Only the Alarm Threshold Month Register is set, and all other Alarm Threshold Registers (Year, Day, Date, Hours, Minutes, Seconds) have their internal bits set to logic 0.

Set the alarm to trigger once per year on March 1 at 12:00:00 AM.
Table 27. Register Contents for the Alarm Trigger Every Year on March 1 @ 12:00:00 AM

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Month | Month | Month | Month | 0 | 0 | 0 | $\begin{aligned} & 10 \\ & M \end{aligned}$ |  | ON |  |  |
| 03h | 03 | 3 (March) | Binary | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

Alarm Example 8 (24-Hour Format): Triggering the Alarm Once a Year Every March 26 @ 23:24:36 Hours

Once every Time, Date, and Month match. Only the Alarm Threshold Month, Date, Hours, Minutes, and Seconds Registers are set, and the other Alarm Threshold Registers (Year, Day) have their internal bits set to logic 0 . Because it is desired for the alarm to trigger each year on the same date, a valid date that exists in each month should be chosen, otherwise the alarm will not trigger during a month that does not contain the particular date (for example, 31 does not exist in June).

Set the alarm to trigger once per year on March 26 @ 23:24:36 hours.
Table 28. Register Contents for the Alarm Trigger Every Year on March 26 @ 23:24:36 Hours

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Seconds | Seconds | Seconds | Seconds | 0 |  | SEC |  | 1SEC |  |  |
| 36h | 36 | 36 | Binary | 0 | 0 | 1 | 1 | 01 | 1 | 0 |
| Minutes | Minutes | Minutes | Minutes | 0 |  | MIN |  | 1MIN |  |  |
| 24h | 24 | 24 | Binary | 0 | 0 | 1 | 0 | 01 | 0 | 0 |
| Hour (24HR Format) | Hour (24HR Format) | Hour (24HR <br> Format) | Hour (24HR <br> Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | 10HR | 10HR | 1HR |  |  |
| A3h | A3 | 23 | Binary | 1 | 0 | 1 | 0 | 00 | 1 | 1 |


| Date | Date | Date | Date | 0 | 0 | 10 | DATE | 1 DATE |  |  |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |
| 26 h | 26 | 26 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 0 |  |  |  |  |  |  |  |  |  |

Alarm Example 8 (12-Hour Format): Triggering the Alarm Once a Year Every March 26 @ 11:24:36 PM

Once every Time, Date, and Month match. Only the Alarm Threshold Month, Date, Hours, Minutes, and Seconds Registers are set, and the other Alarm Threshold Registers (Year, Day) have their internal bits set to logic 0 . Because it is desired for the alarm to trigger each year on the same date, a valid date that exists in each month should be chosen, otherwise the alarm will not trigger during a month that does not contain the particular date (for example, 31 does not exist in June).

Set the alarm to trigger once per year on March 26 @ 11:24:36 hours.
Table 29. Register Contents for the Alarm Trigger Every Year on March 26 @ 11:24:36 PM

| Hex Data | BCD <br> Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Seconds | Seconds | Seconds | Seconds | 0 |  | SEC |  | 1S |  |  |  |
| 36h | 36 | 36 | Binary | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Minutes | Minutes | Minutes | Minutes | 0 |  | MIN |  | 1 MI |  |  |  |
| 24h | 24 | 24 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| Hour <br> (12HR <br> Format) | Hour (12HR Format) | Hour (12HR Format) | Hour (12HR Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | $\begin{aligned} & A / P \\ & (0 / 1) \end{aligned}$ | 10 HR | 1 HR |  |  |  |
| 31h | 31 | 11 PM | Binary | 0 | 0 | 1 | 1 |  | 0 | 0 | 1 |
| Date | Date | Date | Date | 0 | 0 | 10 D | ATE | 1 D | ATE |  |  |
| 26h | 26 | 26 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| Month | Month | Month | Month | 0 | 0 | 0 | 10 M | 1 M | ON | TH |  |
| 03h | 03 | 3 (March) | Binary | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

Alarm Example 9 (24-Hour Format): Triggering the Alarm Once on March 26, 2099 @ 23:24:36 Hours

Once on Time, Date, Month, and Year match. The Alarm Threshold Year, Month, Date, Hours, Minutes, and Seconds Registers are set, and the Alarm Threshold Day Register has its internal bits set to logic 0 .

Set the alarm to trigger on March 26, 2099, @ 23:24:36 hours. This assumes, of course, that the Century Register was programmed for 20 (BCD format).

Table 30. Register Contents for the Alarm Trigger on March 26, 2099, @ 23:24:36 Hours

| Hex Data | $\begin{array}{\|l\|} \hline \text { BCD } \\ \text { Data } \end{array}$ | Decimal Timekeeping Contents | Register/ Binary <br> Data | D7 |  | D5 | D4 | D3 | D2 | D1 | Do |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Seconds | Seconds | Seconds | Seconds | 0 | 10 S | SEC |  | 1SE | EC |  |  |
| 36h | 36 | 36 | Binary | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Minutes | Minutes | Minutes | Minutes | 0 | 10 N | MIN |  | 1 MIN | IN |  |  |
| 24 h | 24 | 24 | Binary | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| Hour (24HR Format) | Hour <br> (24HR <br> Format) | Hour (24HR Format) | Hour <br> (24HR <br> Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | 10HR | 10HR | 1HR |  |  |  |
| A3h | A3 | 23 | Binary | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 |
| Date | Date | Date | Date | 0 | 0 | 10 DA |  |  | DATE |  |  |
| 26h | 26 | 26 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| Month | Month | Month | Month | 0 | 0 | 0 | 10 M | 1 M | MONTH |  |  |
| 03h | 03 | 3 (March) | Binary | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| Day | Day | Day | Day | 0 | 0 | 0 | 0 | 0 | WEE | EKD |  |
| 02h | 02 | 2 (Monday) | Binary | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| Year | Year | Year | Year | 10 YE | AR |  |  |  | YEAR |  |  |

99h 99h 99h Binary 1 |  | 0 | 0 | 1 | 1 | 0 | 0 | 1 |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- | :--- |

Alarm Example 9 (12-Hour Format): Triggering the Alarm Once on March 26, 2099 @ 11:24:36 PM
Once on Time, Date, Month, and Year match. The Alarm Threshold Year, Month, Date, Hours, Minutes, and Seconds Registers are set, and the Alarm Threshold Day Register has its internal bits set to logic 0 .

Set the alarm to trigger on March 26, 2099 @ 11:24:36 PM. This assumes, of course, that the Century Register was programmed for 20 (BCD format).

Table 31. Register Contents for the Alarm Trigger on March 26, 2099, @ 11:24:36 PM

| Hex Data | BCD Data | Decimal Timekeeping Contents | Register/ Binary Data | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Seconds | Seconds | Seconds | Seconds | 0 | 10SEC |  |  | 1SEC |  |  |  |
| 36h | 36 | 36 | Binary | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| Minutes | Minutes | Minutes | Minutes | 0 | 10MIN |  |  | 1MIN |  |  |  |
| 24h | 24 | 24 | Binary | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| Hour <br> (12HR <br> Format) | Hour (12HR Format) | Hour (12HR Format) | Hour (12HR Format) | $\begin{aligned} & 12 / 24 \\ & (0 / 1) \end{aligned}$ | 0 | A/P (0/1) | 10HR | 1HR |  |  |  |
| 31h | 31 | 11 PM | Binary | 0 | 0 | 1 | 1 |  | 0 | 0 | 1 |
| Date | Date | Date | Date | 0 | 0 | 10 DATE |  | 1 DATE |  |  |  |
| 26h | 26 | 26 | Binary | 0 | 0 | 1 | 0 |  |  | 1 | 0 |
| Month | Month | Month | Month | 0 | 0 | 0 | 10 M | 1 MONTH |  |  |  |
| 03h | 03 | 3 (March) | Binary | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Day | Day | Day | Day | 0 | 0 | 0 | 0 | WEEKDAY |  |  |  |
| 02h | 02 | 2 (Monday) | Binary | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |


| Year | Year | Year | Year | 10 YEAR |  |  |  | 1 YEAR |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 99h | 99h | 99h | Binary | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

## More Information

MAX6900: QuickView -- Full (PDF) Data Sheet -- Free Samples<br>MAX6901: QuickView -- Full (PDF) Data Sheet -- Free Samples<br>MAX6902: QuickView -- Full (PDF) Data Sheet -- Free Samples<br>MAX6909: QuickView -- Full (PDF) Data Sheet -- Free Samples<br>MAX6910: QuickView -- Full (PDF) Data Sheet -- Free Samples


[^0]:    $\begin{array}{lllllllll} & \\ \text { ALARM } & 1 & 0 & 0 & 1 & 0 & 1 & 0 & R D\end{array}$
    0 YEAR DAY MONTH DATE HR MIN SEC

