ON Semiconductor

Is Now

To learn more about onsemi™, please visit our website at
www.onsemi.com

AND9992

Using the Window Watchdog Timer Function of the NCV97200

ON Semiconductor®

www.onsemi.com

APPLICATION NOTE

Scope

This Application Note describes the Window Watchdog function of the NCV97200, and how to use it.

Window Watchdog

Gross mis−operation of a microprocessor can be detected by a relatively simple monitor called a Watchdog Timer. In its most fundamental form, this Watchdog is a timer which is prevented from timing out if the processor is operating properly – as indicated by the changing of a logic signal received from the processor. If the Watchdog does time out, it alerts the system or directly initiates a re−boot of the microcontroller.

NCV97200 Watchdog Characteristics

- Watchdog Signal Monitor Input Pin
- Watchdog Fault Output Pin
- Watchdog Timer Programming Pin
- Extended Initial Boot Timer
- Short Pulse Period Detection Timer
- Missing Pulse Detection Timer

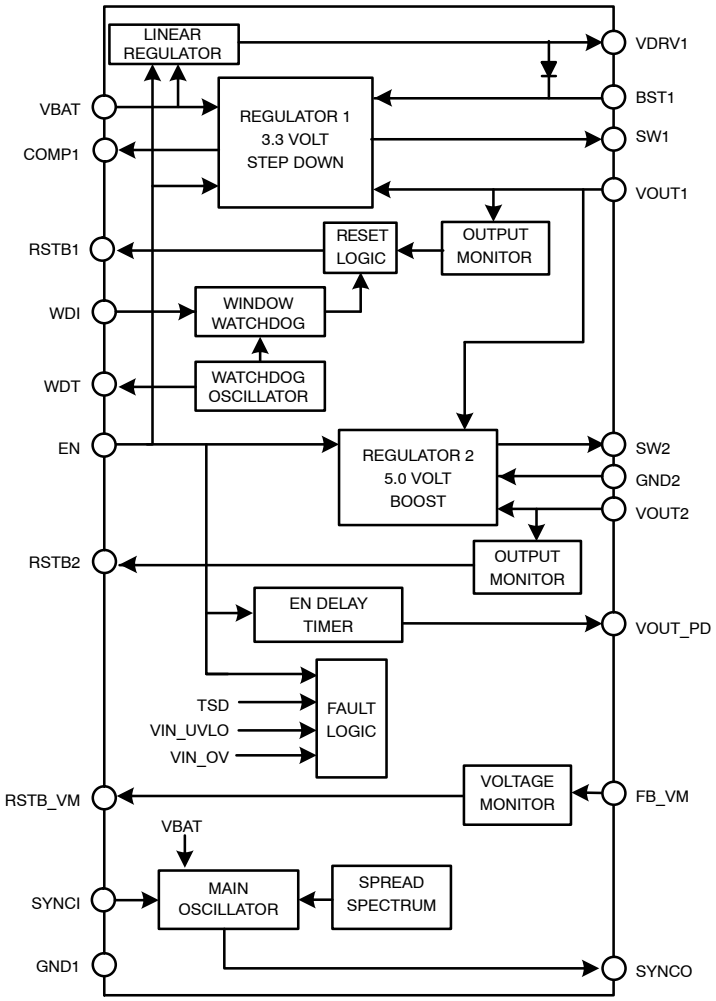

Figure 1. NCV97200 Block Diagram

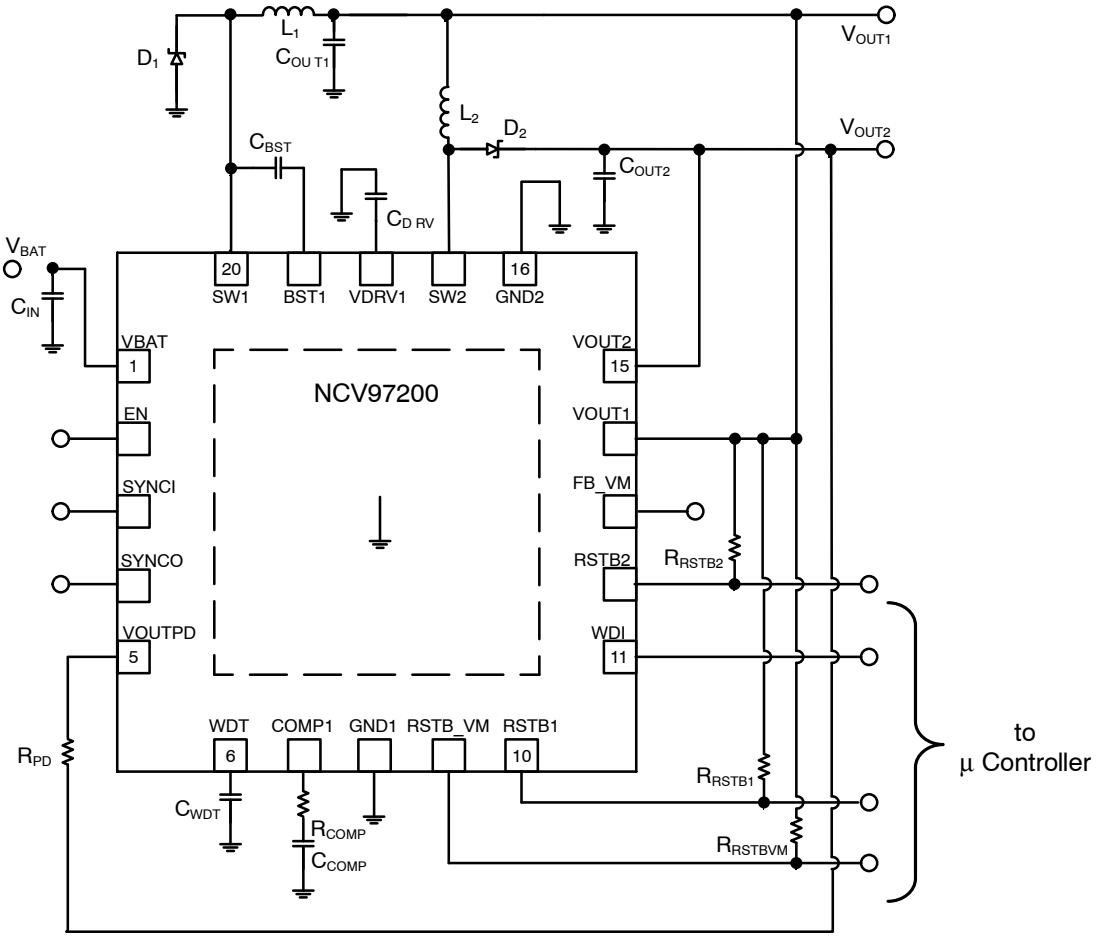

Figure 2. NCV97200 Typical Application

APPLICATIONS INFORMATION

What is the Watchdog Timer?

The NCV97200 Watchdog Timer compares the interval between rising edges of logic pulses at the WDI pin (*Figure 3*) to a set of internal timers. These timers use a dedicated oscillator formed by a current repeatedly charging an external capacitance at the WDT pin. The WDI signal usually comes from a processor (such as a microcontroller) executing a program. Any WDI period within the range set by WDT pin capacitance indicates proper program execution.

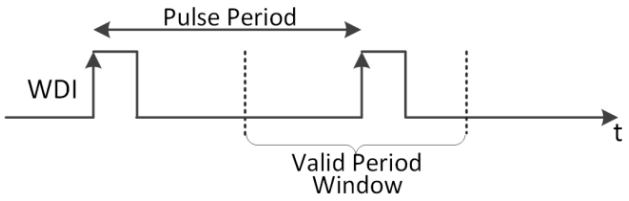

Figure 3. Watchdog Input Signal – WDI

WDI periods outside the valid range (either too long or too short) cause the RSTB1 pin to output a logic low signal in order to reset the processor, and stops the SYNCO output. The NCV97200MW01 outputs keep regulating. The NCV97200MW33 regulators are shut off for the duration of the Reset Delay plus Enable Delay Times.

Watchdog Boot Mode

Before voltage regulator output VOUT1 reaches regulation, the Watchdog is inactive. When active, the Watchdog operates in *Boot Mode* followed by *Window Mode*. *Boot Mode* starts when the NCV97200 sends RSTB1 high – starting the $t_{WD\_timeout}$ timer. RSTB1 high also initiates the processor Boot sequence, which must complete in less time than the $t_{WD\_timeout}$ interval.

Boot Mode lasts until either the end of the $t_{WD\_timeout}$ interval (*Figure 4*), or a rising edge is received at the WDI pin (*Figure 5*). If no WDI edge is received before $t_{WD\_timeout}$ ends, the NCV97200 sends RSTB1 low for the duration of the Reset Delay timer (t_{RESET}) – during which the Watchdog is inactive. When RSTB1 rises, the Watchdog again becomes active in Boot Mode and the $t_{WD\_timeout}$ timer restarts (timeout of one timer starts the other).

The user selects the value of the WDT capacitance to program the $t_{WD\_timeout}$ timer slightly longer than the processor boot−up time. (This programming also determines the interval at which the processor must send Watchdog pulses when in *Window Mode*.)

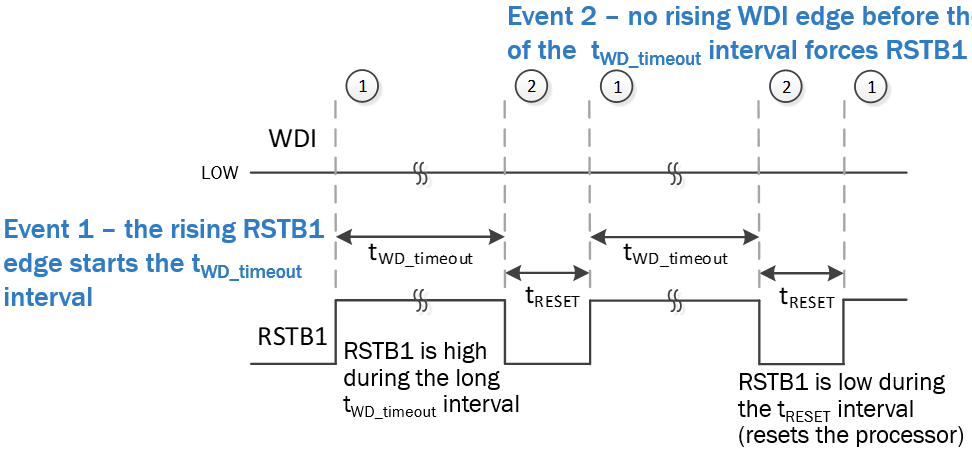

Figure 4. Watchdog Boot Mode Timing

Watchdog Transition to Window Mode

Transition from *Boot Mode* to *Window Mode* occurs when a rising WDI edge occurs during the $t_{WD\_timeout}$ interval (*Figure 5*, event 3). A high level at the WDI pin when the RSTB1 pin goes high is treated as a rising edge received during the $t_{WD\_timeout}$ interval.

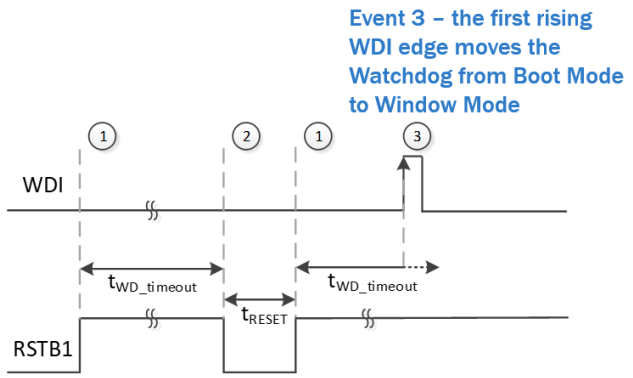

Figure 5. Transition to Window Mode

Watchdog Window Mode

In *Window Mode* (*Figure 6*), the $t_{WD\_timeout}$ timer is replaced by the much (15x) shorter t_{WD} timer – which starts when a rising WDI edge is received. A second, even shorter (4x shorter) timer – t_{WD_CLS} – starts at the same time.

To stay in *Window Mode*, a single rising WDI edge must occur *after timeout of* t_{WD_CLS} *and before timeout of* t_{WD}. Receipt of such an edge immediately restarts both the t_{WD_CLS} and t_{WD} timers.

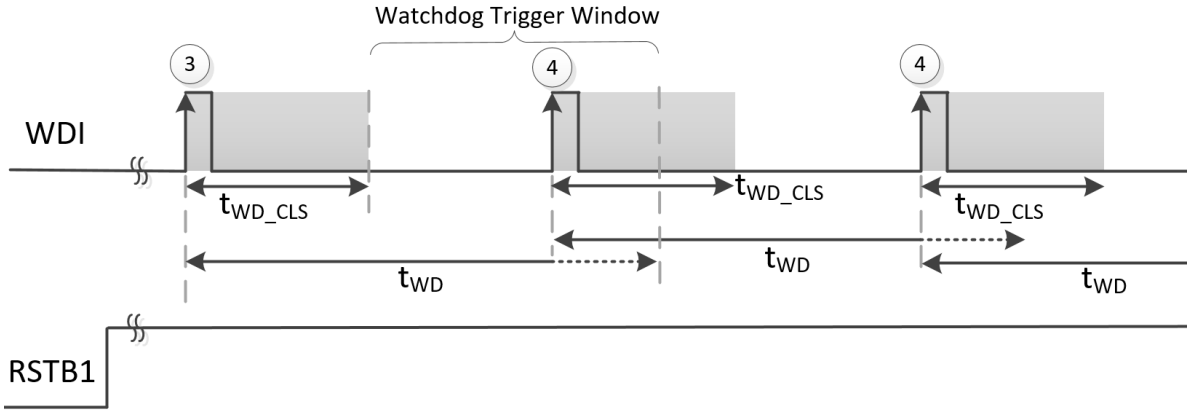

Figure 6. Watchdog Window Mode Timing

Watchdog Transition to Boot Mode

Transition from *Window Mode* to *Boot Mode* can occur for 2 reasons: no rising WDI edge occurs during the t_{WD} interval (*Figure 7*, event 5) or a rising WDI edge occurs during the t_{WD_CLS} interval (*Figure 8*, event 6).

No receipt of a rising WDI edge before timeout of t_{WD}, or receipt of a rising WDI edge before timeout of t_{WD_CLS} indicates abnormal processor operation, and RSTB1 immediately goes low for the Reset Delay Time (t_{RESET}) in order to reset the processor. After the rise of RSTB1, *the Watchdog reverts to Boot Mode* – starting the $t_{WD\_timeout}$ timer in order to allow the processor time to re−boot.

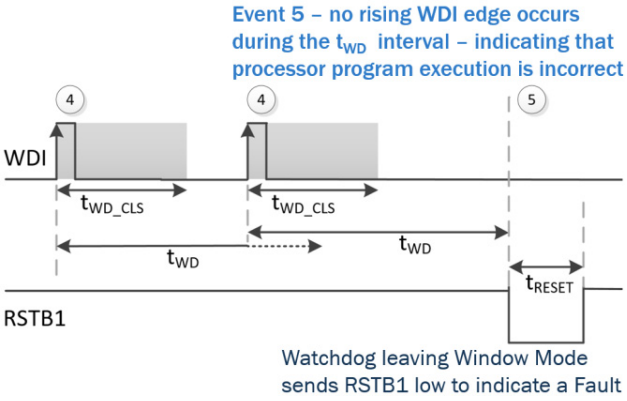

Figure 7. Transition to Boot Mode

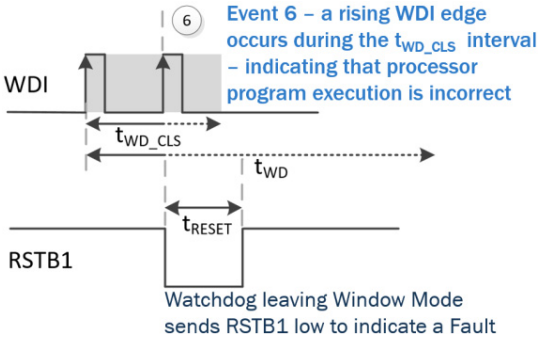

Figure 8. Transition to Boot Mode

Setting Watchdog Timing

PCB−mounted capacitance connected to the WDT pin (C_{WDT}) programs the Watchdog oscillator, which controls the Watchdog timers. This oscillator is independent of the power converter switching frequency oscillator.

Watchdog timing intervals $t_{WD\_timeout}$, t_{WD_CLS} and t_{WD} (but not t_{RESET}) are proportional to C_{WDT} for values between 100 pF and 1000 pF (*Figure 9*). Values above 1000 pF may be used, however linear scaling may not accurately predict times for values above 1000 pF, and the times resulting from values above 1000 pF may exhibit a significant temperature dependence.

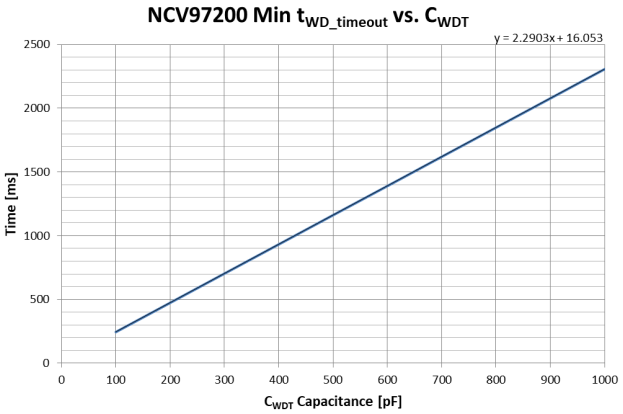

Figure 9. $t_{WD\_timeout}$ versus C_{WDT} Value

Determining the C_{WDT} value

Determine the worst−case maximum processor Boot−up time that could be required under all conditions. This will be the value of the term $t_{BOOTmax}$ in *Equation 1*. $t_{WD\_timeout}$ must be programmed to exceed $t_{BOOTmax}$ in order to ensure that there is enough time for worst−case normal processor boot−up. *Equation 1* calculates the WDT pin capacitance (C_{WDTmin}) that sets the worst case minimum $t_{WD\_timeout}$ equal to $t_{BOOTmax}$ (see *Figure 10*) – including NCV97200 temperature & tolerance effects.

$$C_{WDTmin}\ (pF) = 0.4366 \times t_{BOOTmax}\ (ms) - 7 \qquad \text{(eq. 1)}$$

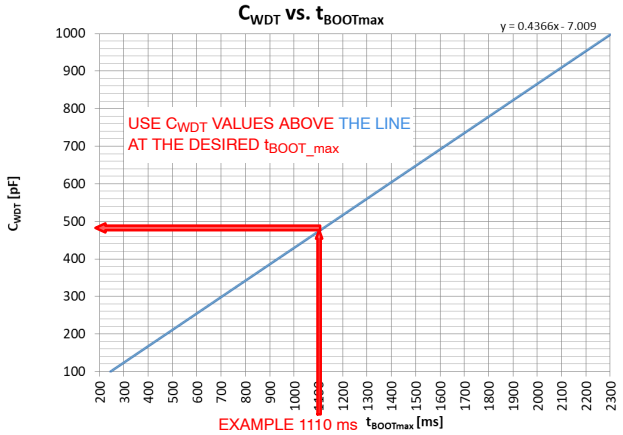

Figure 10. Exact C_{WDT} Value versus $t_{BOOTmax}$

Use *equation 2* to find the lowest standard capacitor value (C_{WDTstd}). [tol = % tolerance of the chosen capacitor plus % variation due to temperature]

$$C_{WDTstd}\ (pF) \geq \frac{C_{WDTmin}\ (pF)}{(100\% - tol)} \qquad \text{(eq. 2)}$$

Equations 3 – 5 calculate the resulting $t_{WD\_timeout}$ range including all temperature & tolerance effects.

$$\text{Minimum } t_{WD\_timeout}\ (ms) = 2.29 \times (100\% + tol) \times C_{WDTstd}\ (pF) + 16 \qquad \text{(eq. 3)}$$

$$\text{Typical } t_{WD\_timeout}\ (ms) = 2.87 \times C_{WDTstd}\ (pF) + 20 \qquad \text{(eq. 4)}$$

$$\text{Maximum } t_{WD\_timeout}\ (ms) = 3.63 \times (100\% + tol) \times C_{WDTstd}\ (pF) + 26 \qquad \text{(eq. 5)}$$

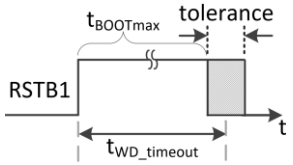

Figure 11. Toleranced $t_{WD\_timeout}$ Interval

Determining the Range of Valid Watchdog Periods

Equations 6 & 7 find the worst−case maximum t_{WD_CLS} and minimum t_{WD} intervals produced by the chosen C_{WDTstd}.

$$\text{Maximum } t_{WD_{CLS}}\ (ms) = \frac{\max\ t_{WDTtimeout}\ (ms)}{60} \qquad \text{(eq. 6)}$$

$$\text{Minimum } t_{WD}\ (ms) = \frac{\min\ t_{WDTtimeout}\ (ms)}{15} \qquad \text{(eq. 7)}$$

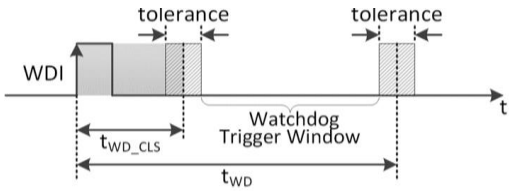

Figure 12. Toleranced Window between t_{WD_CLS} and t_{WD}

These are also the worst−case limits for valid processor−generated Watchdog pulse periods T_{WDI} :

$$\frac{\max\ t_{WDTtimeout}\ (ms)}{60} \leq T_{WDI}\ (ms) \leq \frac{\min\ t_{WDTtimeout}\ (ms)}{15} \qquad \text{(eq. 8)}$$

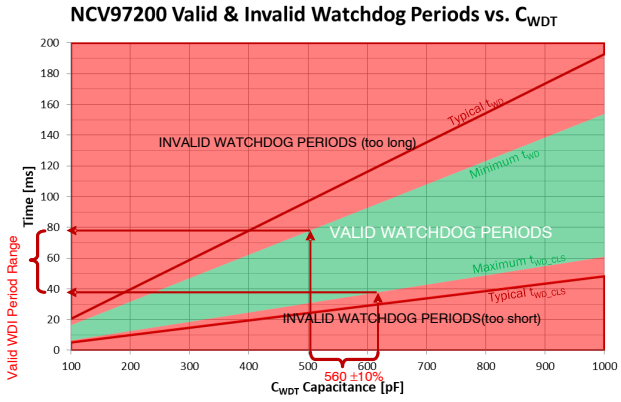

Figure 13. Valid Watchdog Period versus C_{WDT} Value

Example

Per equation 3, the 100 pF, 5% value (populated as C_{WD} on the NCV97200GEVB Evaluation Board) will produce a minimum $t_{WD\_timeout}$ (ms) =
2.29 x (100% − 5%) x 100 + 16 = 233 ms

This capacitor would be suitable for a processor with a worst−case maximum boot time less than 233 ms.

Per equation 5, maximum $t_{WD\_timeout}$ (ms) =
3.63 x (100% + 5%) x 100 + 26 = 407 ms

For this capacitor, the interval given by equation 8 between WDI pulses sent by the processor must be:

$$\frac{407}{60} \le T_{WDI}\ (ms) \le \frac{233}{15}$$

or

$$6.78 \le T_{WDI}\ (ms) \le 15.57$$

An *Excel spreadsheet* implementing these formulas is available on the ON Semiconductor website under NCV97200 Design and Development tools. https://www.onsemi.com/pub/Collateral/NCV97200%20WATCHDOG%20TIMING%20CALCULATOR.XLSX.

Setting Processor Reset Time

The Reset Delay Time (t_{RESET}) is the amount of time the RSTB1 output remains low *after removal of a Fault*. Internally, indication of a Watchdog fault persists for only 3 cycles of the WDT pin oscillator. Since the WDT oscillator period is very short (500 times shorter than t_{WD_CLS}), reliably achieving processor reset (see *Figures 7 & 8*) may require programming t_{RESET} to be longer than minimum.

t_{RESET} is programmed by the amount of current (I_{RSTB1}) flowing through the RSTB1 pullup resistor (R_{PULLUP}) when RSTB1 is low. An R_{PULLUP} producing I_mRSTB1 ≥ 1000 μA sets tμRESET to 1 μs. Much longer delays can be programmed with 100 μA ≤ I_{RSTB1} ≤ 500 μA. (500 μA < I_{RSTB1} < 1000 μA, and I_{RSTB1} > 2 mA should be avoided.)

Determining the RSTB1 Pullup Resistor Value

Determine the worst−case maximum time under all conditions that the RSTB1 output is required to be low in order to reset the processor. This will be the definition of the term $t_{PROC\_RESETmax}$ in the following equations. t_{RESET} must be programmed to exceed $t_{PROC\_RESETmax}$ in order to ensure that the processor is reset. Use *equations 9 & 10* to calculate the R_{PULLUP} value and pullup current which guarantees that t_{RESET} equals or exceeds $t_{PROC\_RESETmax}$ when pulling up to 3.3 V.

$$R_{PULLUP}\ (k\Omega) = 1.737 \times t_{PROC\_RESETmax}\ (ms) \quad \text{(eq. 9)}$$

$$I_{RSTB1}\ (\mu A) = 1000 \times 3.3 / R_{PULLUP}\ (k\Omega) \quad \text{(eq. 10)}$$

Typical Reset Delay time is given by *equation 11*.

$$t_{RESET}\ (ms) = 2475 / I_{RSTB1}\ (\mu A) \quad \text{(eq. 11)}$$

Conclusions

Programming the NCV97200 to reliably monitor and properly respond to improper microprocessor execution requires knowledge of the maximum processor boot and reset times. That knowledge, along with the procedures and equations in this Application Note, provide the means to determine the correct standard value for the Watchdog timing capacitor and the minimum value of the Reset Delay programming resistor.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:
Email Requests to: orderlit@onsemi.com

ON Semiconductor Website: www.onsemi.com

TECHNICAL SUPPORT
North American Technical Support:
Voice Mail: 1 800−282−9855 Toll Free USA/Canada
Phone: 011 421 33 790 2910

Europe, Middle East and Africa Technical Support:
Phone: 00421 33 790 2910
For additional information, please contact your local Sales Representative