



APPLICATION NOTE 3932

Using the MAXQ3210 as a Microprocessor Supervisor

Abstract: The MAXQ3210 microcontroller can perform a variety of microprocessor supervision and support tasks including power-on reset control, power sequencing, watchdog timeout, and input switch debouncing. Depending on the application, the MAXQ3210 can also supply power to the microprocessor directly. This application note shows how the MAXQ3210 runs directly from a 9V battery, and how with no external clock-crystal components it performs several functions normally supplied by a microprocessor supervisor for low-speed applications.

Overview

When microprocessors are integrated into a larger design, they often require the support of a variety of external components. These components perform several tasks for the microprocessor: supply regulated power at the voltages and current capabilities needed by the microprocessor; monitor or control other power-supply regulators; provide an external clock; control power-up and power-down sequencing; or perform a number of other support tasks. Often these components, generally referred to as microprocessor supervisors, are dedicated ICs designed for a particular supervisory task.

Instead of dedicated microprocessor supervisors, one can use a small, low-power microcontroller for these various support purposes. Depending on its capabilities, a microcontroller can supply the support services of several components, while adding flexibility since the supervisor functions are implemented in software.

This application note describes ways to perform typical microprocessor supervisory functions using the low-power [MAXQ3210](#) microcontroller with piezoelectric horn driver, voltage regulator, and comparator. The MAXQ3210 performs many of these functions in a completely stand-alone configuration; it can operate from a 9V battery and does not require an external clock circuit when using its internal low-speed ring oscillator. The small package size (24-pin TSSOP) of the MAXQ3210 is also an advantage when fitting it into legacy designs or other systems where space is at a premium.

Battery Power for Portable Systems

The MAXQ3210 contains an integrated 9V to 5V linear regulator. This makes the device an ideal basis for a portable, battery-powered microcontroller or microprocessor system. Since the MAXQ3210 also provides a low-battery supply monitor, the system can react intelligently to a low-battery state.

When supporting a 5V microprocessor, the MAXQ3210 can power the microprocessor directly from its voltage-regulator output pin, REGOUT (**Figure 1**), as long as the supply current required by the microprocessor is within the limits of the MAXQ3210's linear regulator (approximately 50mA max. Refer to the MAXQ3210 data sheet for more details.). Microprocessors with extremely low-current requirements can be powered directly from one of the MAXQ3210's port pins set to standard high-output mode (PD = 1, PO = 1). Each of these port pins provides a high-output current of at least 4mA with a voltage drop of no more than 0.4V. Multiple port pins can be tied together to increase the amount of supply current, as long as the maximum rating for combined current on all port pins is not violated (approximately 25mA. Refer to the data sheet for more details). Supplying the microprocessor directly from a port pin(s) allows the MAXQ3210 to power the microprocessor on and off completely as needed to conserve battery life. **Note:** when using REGOUT to supply the microprocessor, power can still be cycled on and off under MAXQ3210 control, but an external FET or other switch component is required.

When supporting a microprocessor which requires a supply rail other than 5V (e.g., 3V, 2.5V, or other), the

MAXQ3210's REGOUT supply can power a linear regulator which generates the supply rail for the microprocessor. By using a linear regulator with an enable/disable input, the MAXQ3210 still shuts the microprocessor power supply off completely to conserve battery life.

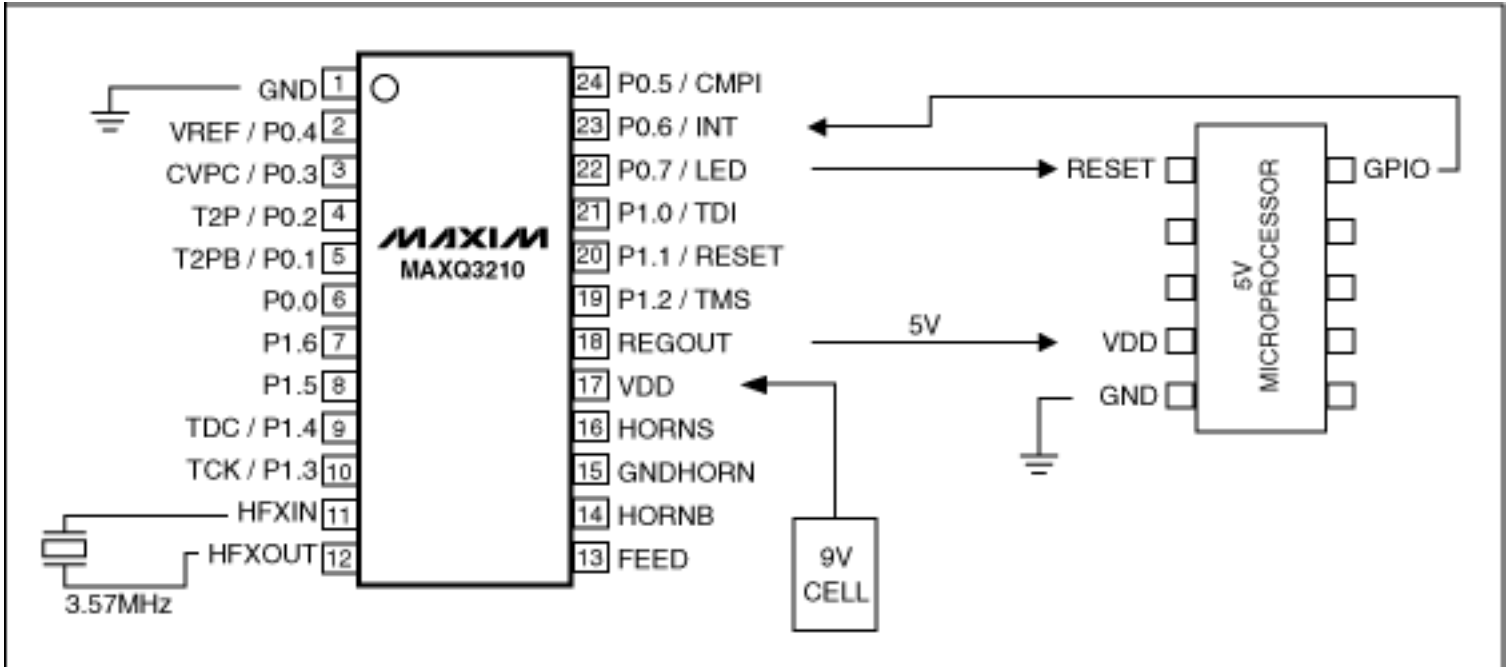


Figure 1. Schematic illustrates how to supply power to a 5V microprocessor from the MAXQ3210.

When battery life and overall current drawn by the system is a design concern, the MAXQ3210 can operate in an extremely low-current state to conserve power. Not only are the full range of MAXQ® power management modes (clock divide, PMM1 mode, and Stop mode) available to conserve power, but the MAXQ3210 also contains an 8kHz ring oscillator that can be used as an extremely low-power clock source. When no external crystal or high-frequency clock source is provided upon powerup, the MAXQ3210 will run exclusively from the 8kHz ring oscillator. This action reduces its operating current to the microamp range. All the supervisory functions described in this application note are easily performed by the MAXQ3210 even when running at 8kHz, since rapid response time is not a requirement.

Reset and Supply Monitoring

The MAXQ3210 provides integrated power-on-reset (POR) and brown-out detection, which allow it to wait until its power supply has stabilized before exiting the reset state. The brown-out detection feature enables the microcontroller to recover cleanly from power-supply disruptions. The MAXQ3210 also provides the same type of POR and brown-out detection to microcontrollers lacking these power-supply monitoring features.

Connecting a port pin from the MAXQ3210 to the microprocessor's external RESET pin allows the MAXQ3210 to reset the microprocessor when its power supply goes below an acceptable level, and release the microprocessor from reset when its power supply recovers. If the microprocessor runs from a 5V supply, the MAXQ3210 simply resets the microprocessor whenever the MAXQ3210 experiences a POR or brown-out condition. If the microprocessor runs from a lower supply rail (such as 3.3V), the MAXQ3210's analog comparator can compare the microprocessor's power supply to an arbitrary level (up to approximately 3.5V) provided at the VREF pin. (Refer to the MAXQ3210 data sheet for analog comparator details.) When the microprocessor's power supply drops below the reference level, the comparator generates an interrupt to the MAXQ3210, which responds by holding the microprocessor in reset until the power supply recovers.

Since all of these functions are implemented in software, the microprocessor's interval between power-supply recovery and release from reset can be set to any time period. If there are multiple power supplies or components in the system which need to be powered on in a particular order, the MAXQ3210 acts as a power-supply sequencer, turning on components and power supplies in a given order with specified delays between each action.

An external reset pushbutton added to the system will initiate a reset sequence. Connecting the reset

pushbutton to one of the MAXQ3210's port pins allows it to perform software-based debouncing functions and to generate a programmable-length reset pulse in response to a button press.

Adding a Wakeup Timer

In portable battery-powered systems, conserving power and battery life is a key design concern. To keep current consumption low, these types of systems typically go into a low-power state whenever possible. A system might wake up periodically (or at a user's request through a button press), take some measurements or perform another activity, and then go back to sleep. Components not in use are powered off as much of the time as possible.

The MAXQ3210 provides a long period wakeup timer to assist in this type of power conservation. The wakeup timer, which can be set to run from the 8kHz ring oscillator, allows the MAXQ3210 to go into its lowest power consumption state—Stop mode—and wake up automatically after a predefined interval as long as two minutes. After exiting Stop mode, the MAXQ3210 wakes up or powers on all other components in the system as needed. The following MAX3210 code demonstrates how to go into Stop mode and automatically wake up again after 60 seconds.

```
mainLoop:
    ;; Perform application tasks here...
    ;;

;; Start the wakeup timer for 60 seconds.

    move    CKCN.6, #1          ; Select ring oscillator mode
waitRing:
    move    C, CKCN.5
    jump    NC, waitRing       ; Wait for RGMD=1 (running from ring)

    move    WUT, #30000        ; 1/8kHz * 30000 * 16 = 60 seconds
    move    WUTC, #0101b       ; Start the wakeup timer (running from ring)

    move    IV, #wakeUpInt     ; Set interrupt handler for wakeup interrupt
    move    IMR.0, #1          ; Enable interrupts from module 0
    move    IC.0, #1           ; Globally enable interrupts

    move    CKCN.4, #1         ; Go into Stop mode, wait for wakeup int
    nop

    jump    mainLoop           ; Back for another round

;;;;;;;;;;;;;;

wakeUpInt:
    move    CKCN.6, #1          ; Select ring oscillator mode
wakeUp_ring:
    move    C, CKCN.5
    jump    NC, wakeUp_ring     ; Wait for RGMD=1 (running from ring)

    move    WUTC, #0           ; Clear wakeup timer flag

    move    CKCN.6, #0         ; Select crystal mode
wakeUp_xtal:
    move    C, CKCN.5
    jump    C, wakeUp_xtal     ; Wait for RGMD=0 (running from crystal)

    reti
```

The microprocessor signals the MAXQ3210 that it is ready to be powered down by using a connection from one of its port pins to a port pin on the MAXQ3210. Immediately after signalling the MAXQ3210 to shut it down, the microprocessor should either go into its own low-power Stop mode or wait to be hard powered off, depending on the implementation.

When the MAXQ3210 wakes up from Stop mode, it either powers the microprocessor on (if it was shut off completely), or uses an external interrupt or similar mechanism to signal the microprocessor to wake up from its low-power state.

Adding a Watchdog Reset

As with the MAXQ3210's wakeup timer, the microcontroller's built-in watchdog reset can be extended as an external hardware watchdog reset for the microprocessor. On the MAXQ3210, the watchdog reset acts as a safeguard against momentary glitches which might lead to runaway code. Once the watchdog timer is armed, it must be reset periodically by setting a register bit. If the MAXQ3210 stops setting the register bit and the timer runs out, the watchdog causes the MAXQ3210 to automatically reset. This automatic reset allows it to recover and begin again from a known starting state.

Extending the watchdog reset from the MAXQ3210 to the microprocessor requires only two connections: one from a port pin on the microcontroller to the microprocessor's external RESET pin, and one from a microprocessor port pin back to the MAXQ3210. The MAXQ3210 enables the watchdog automatically, and resets it whenever it receives a signal from the microprocessor's port pin. The microprocessor's responsibility is to toggle this pin often enough to prevent the MAXQ3210's watchdog from timing out. The timeout period is programmable. (Refer to the [MAXQ3210 User's Guide](#) Supplement for more details.)

As an external watchdog for the microprocessor, the MAXQ3210 sets its watchdog to trigger an interrupt when it times out. If the watchdog times out, that means that the microprocessor failed to toggle the microcontroller's port pin in time, and that the microcontroller needs to be reset. The MAXQ3210 responds to the watchdog interrupt by restarting the watchdog, resetting the microprocessor by means of its external RESET pin, and beginning the cycle again.

Levels and Interfacing

When running signals between the MAXQ3210 and the microprocessor under supervision, the respective I/O voltage levels of the two devices must be considered carefully. If both devices run at 5V, port pins generally connect directly with no adjustment needed. Always check the respective data sheets to be certain.

Output signals running from the MAXQ3210 to a lower voltage microprocessor (such as a 3.6V device) are also straightforward to implement. If the microprocessor's pins contain protection diodes, the 5V outputs from the MAXQ3210 connect directly to the lower voltage pins through a current-limiting resistor. An alternative is to use an open-drain connection method: connect a pullup resistor from the two pins to the lower voltage microprocessor supply and drive the MAXQ3210 port pin in either strong pulldown (PD = 1, PO = 0) or tristate (PD = 0, PO = 0) mode.

Inputs on the MAXQ3210 which are driven by port pins on the lower voltage microprocessor require further consideration. This is because the lower voltage microprocessor cannot drive its port pins to a high enough level to register as a logic 1 on the MAXQ3210. If only one input is needed (as in Figure 1), the low-voltage signal can be connected to the analog comparator on the MAXQ3210. Other possibilities include using an open-drain configuration with an external pulldown FET, or a discrete level shifter such as the MAX3370.

Refer to application note 3007, [Logic Level Translation](#) for an additional discussion of this issue.

Conclusion

The small size, battery-power support features, and stand-alone operation mode of the MAXQ3210 make it an excellent choice for a programmable microprocessor supervisor. With a number of useful power-conservation and supply-monitoring features that can easily be extended to the microprocessor to be supervised, the

MAXQ3210 forms the core of a portable, battery-powered system that can be configured using software for optimal battery life.

MAXQ is a registered trademark of Maxim Integrated Products, Inc.

Application Note 3932: www.maxim-ic.com/an3932

More Information

For technical support: www.maxim-ic.com/support

For samples: www.maxim-ic.com/samples

Other questions and comments: www.maxim-ic.com/contact

Automatic Updates

Would you like to be automatically notified when new application notes are published in your areas of interest?

[Sign up for EE-Mail™.](#)

Related Parts

MAXQ3210: [QuickView](#) -- [Full \(PDF\) Data Sheet](#) -- [Free Samples](#)

AN3932, AN 3932, APP3932, Appnote3932, Appnote 3932

Copyright © by Maxim Integrated Products

Additional legal notices: www.maxim-ic.com/legal