BY MATT RUFF • FREESCALE SEMICONDUCTOR

Reducing power consumption in battery-powered applications

USING THE LOWER POWER MODES OF ADVANCED MICROCONTROLLERS MEANS LESS BATTERY DRAIN IN PARKED AUTOMOBILES AND LONGER BATTERY LIFE IN PORTABLE CONSUMER PRODUCTS.

Automotive-electronics content is expanding at an ever-increasing pace. Today, an average car has 25 to 35 microcontrollers; approximately half of those are 8-bit units. High-end vehicles can easily have 80 or more microcontrollers. With rising energy costs fueling an explosion of hybrid technologies and car buyers demanding more comfort features and conveniences, automakers look to electronics for the solutions. This search translates to a dramatic rise in the need for computing power under the hood and in the cabin. The system challenge that all that computing power creates is a greater demand for electrical power. Because increasing the electrical capacity of the car generally adds weight and reduces fuel economy, the alternative is to either use less power or make more efficient use of the available power in the car.

In cars, an alternator supplies 100A or more to handle the loads, so saving a few microamps may seem unimportant. In fact, the biggest concern for saving power in the automotive market is for key-off modes, which stay on when the ignition key is off and the car is parked. In many cases, the microcontroller is the largest consumer of power in the key-off mode, rivaled only by wireless communications, such as the transmitter/receiver. The power consumption is a function of how frequently you power up the microcontroller and what type of operation it is performing. These microcontrollers must wake up periodically from a sleep mode based on a very-low-power oscillator. They quickly come up to operation, perform a required function, and then go back to sleep. A good example is a keyless-entry system in the body controller. This system wakes up periodically to see whether the key fob has sent a signal to open the doors.

In the key-off mode, automotive and consumer-electronics applications are similar in that they are both battery-powered. In portable electronics, power consumption is a major design consideration because it directly impacts battery life and the useful operation of the product. Fortunately, with the latest microcontrollers, designers can reduce power consumption, and the same principles apply to any power-conscious application.

Historically, at the module level, the maximum current draw under a key-off situation was 1 mA. With the increasing number of modules and loads that are active during the key-off mode, most carmakers are implementing or considering a maximum current draw of a fraction of this level. For example, as **Table 1** shows, at Volkswagen, the initial Golf had three modules (**Reference 1**). The number increased to five, 11, 18, and, finally, 25 in the current Golf V model. The **table** also shows the corresponding changes in total vehicle sleep current.

Many of the largest gains in efficiency will be in comfort and convenience systems, in which the predominant hardware architecture is the 8-bit microcontroller. These systems have a

TABLE 1 VOLKSWAGEN GOLF TRENDS

Volkswagen model	No. of modules	Total vehicle sleep current (mA)	Sleep current per module (mA)
Golf I	Three	8.3	2.766
Golf II	Five	10.5	2.1
Golf III	11	16.2	1.473
Golf IV	18	21.3	1.183
Golf V	25	15.4	0.618

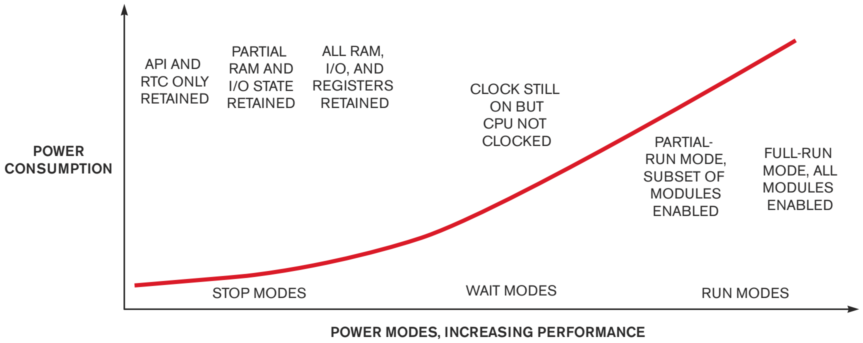

Figure 1 Flexible low-power modes allow progressively lower power consumption as active circuitry's performance decreases.

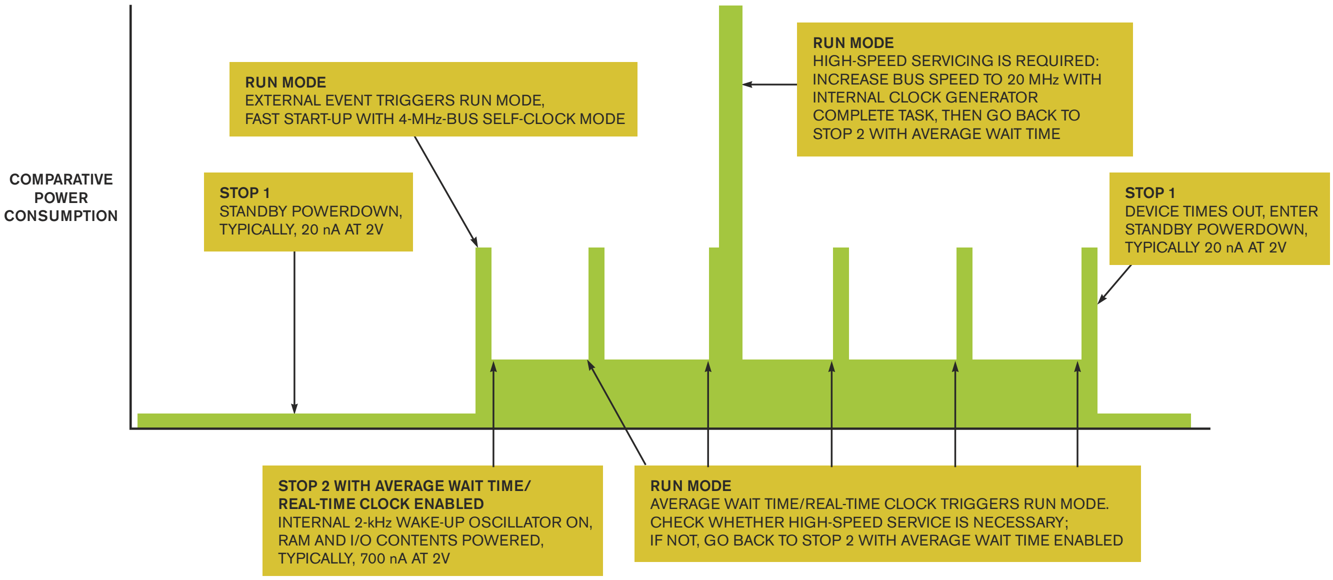

Figure 2 The power-consumption profile for stop and run modes includes a periodic wake-up.

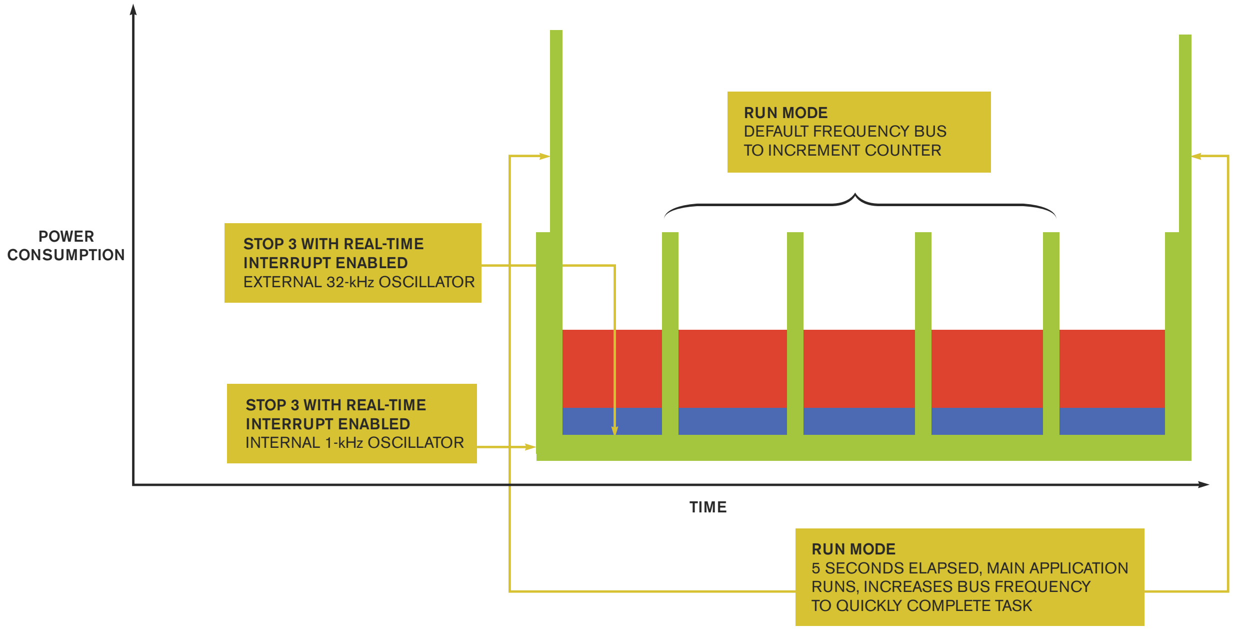

Figure 3 Extend battery life with clock management by implementing periodic wake-up and using a lower clock frequency in the run mode to reduce the average power consumption.

relatively large number of electronic components, and automakers continue to add more comfort and convenience features to a wider number of vehicle platforms. Depending on the design, the microcontroller can be a significant or a small part of the problem. With some of the latest microcontroller considerations for power, the microcontroller can become a part of the solution and provide the means to satisfy lower current requirements. This approach requires implementing the power-saving features of the next generation of 8-bit microcontrollers.

DECREASING POWER CONSUMPTION

Highly customizable features, such as variable clock speeds and multistage power-down modes, allow system designers to optimize microcontroller performance to suit not only the computing requirements of the application, but also the increasingly important power budget. More advanced clock generation also allows the user to choose how the microcontroller generates clocks to maximize the performance for a given power-consumption level. Periodic wake-up capabilities and the ability to wake up the microcontroller based on network communications or other outside signals allow the device to remain switched off until a user needs it, further extending power savings. These low-power options couple with system-protection features, such as low-voltage detection, which ensure that the application continues to operate safely while running on minimal power. These advanced capabilities hold the key to ever-more-efficient automotive-electronic systems and extend the electronic-content capacity of tomorrow's vehicles.

The traditional high-level power strategies still apply. First, turn off power-consuming circuitry whenever possible, and, second, minimize the operating frequency of the system under all conditions. The newest microcontroller designs include more options for the system designer to reduce power consumption in many of the available modes. Specific modes allow the system designer to reduce power whenever possible by switching on only the necessary portions of silicon.

The modes include several stop or deep-sleep states; an intermediate wait or doze state; a modified run mode, which is a subset of microcontroller awake; and run mode, when the chip is fully awake (**Table 2**). Consumer and industrial applications use the Stop 1 mode, which is not available for automotive microcontrollers that operate at 5V. The consumer and industrial applications typically have lower operating voltages (**Reference 2**). The lowest power-consumption mode, Stop 1, is basically a shutdown mode, with a typical drain-to-drain current as low as 20 nA at 2V. This mode leaves only active circuitry, such as an interrupt pin or a pin input, powered to allow the unit to wake up. In this state, the microcontroller wakes up on an external event, edge, or level trigger. The trade-offs of this mode are that is loses RAM content, it resets all register contents, and it puts the I/O pins in the reset state.

In a partial power-down mode, such as Stop 2 with a drain-to-drain current as low as 400 nA at 2V, the microcontroller wakes up with an IRQ (interrupt request), reset, or internal RTI (real-time interrupt) but does not lose the RAM contents, and the I/O states are latched. The RTI can wake up the microcontroller without the occurrence of an external event. One limitation is that the register values are reset; however, you can save those values to RAM and restore them.

Another partial power-down mode, such as Stop 3 with a typical drain-to-drain current as low as 500 nA at 2V, is comparable with earlier microcontroller stop modes in that wake-up occurs with any active interrupt, including IRQ, KBI (keyboard interrupt), LVD (low-voltage detection), RTI, or reset. In addition to retaining RAM and register values without requiring initialization of peripherals, the system can use an external clock as a high-accuracy input into the RTI. The trade-off for this mode is higher current draw.

Because the CPU is the most power-hungry block on the microcontroller, you should switch it off as often as possible. To increase the number of times and situations in which you can turn them off, microcontrollers may employ a number of autonomous peripherals that require no CPU to operate. Key autonomous peripherals include the RTI, which allows the microcontroller to recover from a very-low-power state at selectable time intervals, and an RTC (real-time clock), which allows timekeeping functions in a very-low-power state. In addition, the ADC can perform continuous conversion while running in low power and trigger a wake-up when the signal reaches a user-defined level.

The RTI and ADC are parts of the fast-wake-up mechanisms that allow the microcontroller to more frequently enter stop modes at the lowest power-consumption level. These blocks increase the speed at which the device can recover from low-power modes and start execution. Once a clock is present, reconfiguration and non-timing-critical operations can start. However, very fast wake-up requires an on-chip IRC (internal reference clock), which enables almost immediate operation.

A variety of low-power-clock sources provide system designers options in start-up time and accuracy for lowering power consumption depending on the system's performance requirements (**Table 3**). The ability to implement variable clocking allows throttling back the frequency for lower current and powering up the clock rate to full speed when math-intensive computing is required.

With the new power-savings tools, designers have greater flexibility in making power-versus-performance decisions

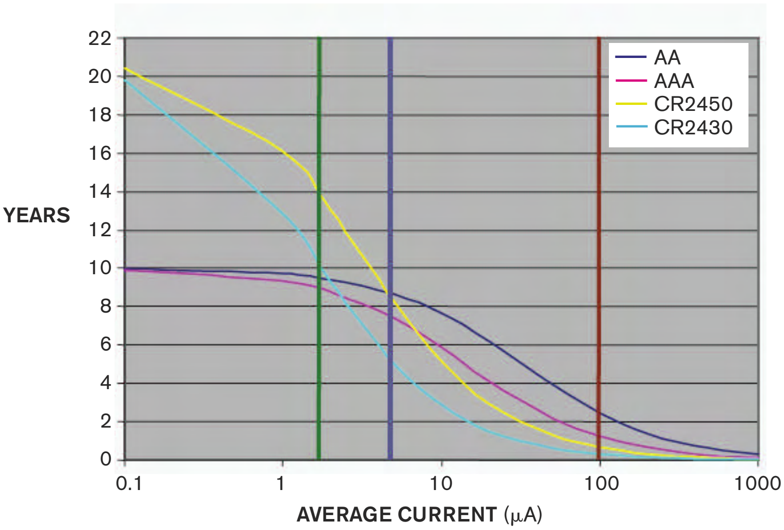

Figure 4 Reducing the average current draw has different impacts on battery life. The lowest levels of stop-, wait-, and run-mode current maximize battery life to meet 10-year operating-life goals.

TABLE 2 STOP MODES

Mode	CPU, digital peripherals, flash	RAM	Interactive computer graphics	Arrival-time difference	Keyboard interrupt	Regulator	I/O pins	Real-time interrupt
Stop 1	Off	Off	Off	Disabled	Off	Off	Reset	Off
Stop 2	Off	Standby	Off	Disabled	Off	Standby	States held	Optionally on
Stop 3	Standby	Standby	Standby	Disabled	Optionally on	Standby	States held	Optionally on

(**Figure 1**). The **figure** identifies power modes with increasingly higher power consumption in stop, wait, and run modes. This flexibility leads to lower power consumption with a properly implemented power-management strategy. One of the strategies for reduced power consumption is periodic wake-up, which couples with the power-management modes (**Figure 2**). Even microcontrollers without an extremely low-power mode can extend battery life using a power-management scheme that uses other available stop modes (**Figure 3** and **Reference 3**).

From a systems approach, analog power-management ICs can augment the power savings in the microcontroller portion. Freescale calls these power-management chips SBCs (systems base chips), which have many aspects and variations. The fundamental pieces include voltage regulation; input- and output-drive circuitry; and, frequently, a physical layer for the communications protocol, such as a CAN (controller-area-network) or LIN (local-interconnect-network) transceiver (**Reference 4**). The key aspect is the voltage regulation for the microcontroller with a built-in communication module. With the combination of one analog IC and the microcontroller, you can put the entire system into the sleep mode with the microcontroller's power off and the regulator shut down to save even more power.

TABLE 3 CLOCK SOURCES AND THEIR ATTRIBUTES

Clock source	Start-up time	Power	Accuracy
Internal reference	Less than 100 μsec	Low	Less than 2%
Internal reference with frequency-locked loop	Less than 100 μsec	Low	Less than 2%
External reference	Milliseconds	Medium	Crystal
External reference with frequency-locked loop	Milliseconds	High	Less than 1%
External reference with phase-locked loop	Milliseconds	High	Less than 1%

Network activity turns the regulator back on, which powers up the microcontroller, allowing the system to perform the required operations. This approach saves the most current because it involves advanced power-saving capabilities, and designers can do several tricks to save even more power.

One automotive application with similar requirements to those of consumer products is the TPMS (tire-pressure-monitoring system). With the complete electronic system, including the microcontroller, sensors, transmitter/receiver, and the battery mounted inside the tire, a TPMS works just like any other battery-powered device—except that the user cannot replace the battery. This automotive application and others like it

MORE AT EDN.COM

For additional stories covering power consumption and energy-efficiency benchmarking, visit

www.edn.com/article/CA495734

www.edn.com/article/CA379884

To post a comment on this article, go to www.edn.com/ms4237 and click on Feedback Loop.

demand the maximum effort to reduce microcontroller-power consumption. The rule of thumb in a TPMS is a 10-year battery life. The coin-cell lithium-ion battery in these applications typically has 1100-mAhr capacity, but companies working on these systems want to shrink the battery. Design goals to reduce the battery size to at least half of this level make aggressive power-management efforts even more important. A TPMS could implement as many as four stop modes, a wait mode, and a run mode to reduce power consumption and meet the 10-year target life when the average current is below 2 μA (**Figure 4**).

Because lower power consumption and optimized power management have become integral parts of system design, microcontroller designers will continue to squeeze current levels to provide very-low-power-rated products. In the future, the lowest power-consumption microcontrollers will have even lower current by reducing the contribution from digital circuitry and by using library and process modifications. In addition, reducing the voltage output of the voltage regulator and implementing new low-power run and wait modes will take power consumption to even lower levels. EDN

AUTHOR'S BIOGRAPHY

Matt Ruff is an engineer with Freescale Semiconductor's automotive-systems-engineering organization. He graduated from The University of Texas—Austin in 1997 with a bachelor's degree in electrical engineering. During college, he worked for Alcatel Network Systems in the company's interoperability labs. Ruff began his career with Motorola in February 1997, working as an applications engineer. Throughout the last eight years, he has been part of the systems-engineering team at Motorola and now carrying over to Freescale Semiconductor, specializing in automotive-multiplexing technologies, such as CAN (controller-area network), LIN (local-interconnect network), and J1850. He has helped to develop several multiplexing microcontrollers and holds a patent related to LINs. He has been a member of the Society of Automotive Engineers and a member of the SAE Vehicle Networks for Multiplex and Data Communication Standards Committee for seven years as well as a member of the SAE J2602 LIN Task Force.

REFERENCES

1. "AN2493, MC9S08GB/GT Low-Power Modes," www.freescale.com/files/microcontrollers/doc/app_note/AN2493.pdf .
2. "Using 8-bit microcontrollers in Low Power Applications," www.freescale.com/files/ftf_2006/doc/presentations/Americas/AE104.pdf.
3. "Meeting the Low Power Challenge in the Automotive Body Space," www.freescale.com/files/ftf_2006/doc/presentations/Americas/AA125.pdf.
4. "MC33689: System Base Chip with LIN," www.freescale.com/webapp/sps/site/prod_summary.jsp?code=MC33689&nodeId=01435940585795 8445.