How to Reduce Power Consumption in EV HMIs

by Jeff Stewart and CC Hung

Consumers are acutely aware that in electric vehicles (EVs), range is everything. So is the U.S. government; the infrastructure bill currently moving through Congress includes $7.5 billion for hundreds of thousands of new charging stations across the country, just to keep EVs on the road.

Obviously, every milliwatt-hour of EV battery power counts. Efficiency in power utilization must be the objective of every system, on every vehicle. That extends to the human machine interface (HMI) system in EVs, which spans driver cockpit displays, head-up displays (HUDs) and in-vehicle infotainment (IVI) control panels.

Designers and engineers must be concerned with minimizing power consumption at every level. With the right configuration, HMIs can be up to 50% more efficient—and this is how.

Reviewing Power Utilization

Display-based HMI systems utilize electricity to drive their various components, as shown in the following chart:

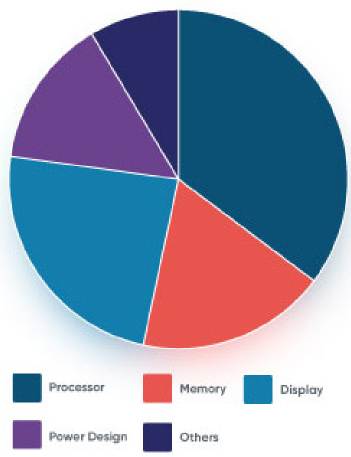

Figure 1: Building Blocks of Power-Efficient System

Video displays, unfortunately, are fixed in their power usage by the backlight technology employed. Dimming the display can reduce power consumption, but this is rarely practical. Processors, on the other hand, draw a lot of attention as candidates for greater efficiency because they are the most power-hungry component.

Processors utilize power to cycle between transistor states and even draw a small amount of power when at rest. Clock frequency, however, is the most telling metric of power consumption in semiconductors. Frequency is defined by state changes as capacitors charge and discharge. Fewer software instructions means the same task can be completed with a lower clock frequency and less power utilization.

Instructions, in turn, are determined by the amount of software code employed. Fewer lines of software mean fewer instructions, the possibility to run lower clock frequency and less electricity consumed.

Memory can be a target for efficiency as well. Each bit of information stored in memory requires power to maintain data states. The more capacity a memory block has, the more power is required to keep it active. In fact, excess memory can drain battery power needlessly. Battery-powered devices pause apps that require high RAM usage when not in use, just to conserve battery power.

The bottom line is that more efficient HMI code—that is, fewer lines of code and less RAM needed to execute a given task—has benefits on both the processor and memory power consumption. Optimizing the HMI software can reduce cockpit HMI system power requirements by 25% or more;

independent testing done by several automotive OEMs has demonstrated RAM, CPU, flash usage is reduced by 50% or more in actual deployments using Altia HMI development software. An approximation of the savings is shown here:

With these facts in mind, why don't EV HMI software designers naturally put code efficiency at the top of their list? Inertia holds a clue.

Many software development teams use automated web-based tools, along with object-oriented languages, to produce code. This simplifies the development process but results in less efficient, bloated code. Tight code, by contrast, takes time. It requires a considerable amount of upfront effort by disciplined, experienced developers (the kind that are in short supply across the industry). The result is software that requires less electricity to operate; the kind of power efficiency that adds up over time.

Improvements are Significant

The mileage savings from a software-efficient HMI is small over a single EV battery charge. Calculations show that the miles lost (20) due to HMI operation and the miles saved by more efficient code (5) over a vehicle's lifetime are not, in and of themselves, significant. But when code efficiency is viewed in the aggregate, the benefits become both apparent and meaningful.

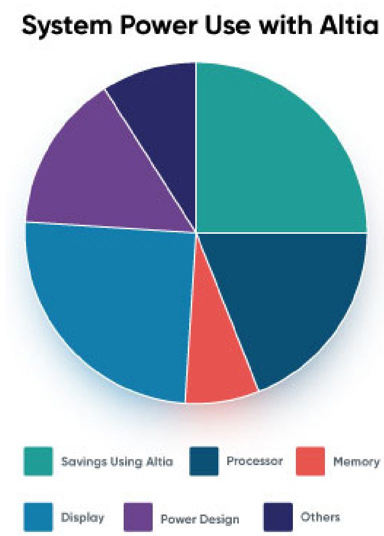

Figure 2: System Power Use with Altia

In terms of CO_2 emissions due to battery charges, over 5.2 million pounds of greenhouse gas is saved over the operational lifetime of an entire EV fleet—the equivalent to the annual output of 429 homes, according to the U.S. EPA's [Greenhouse Gas Equivalencies Calculator].

Put another way, for a fleet of one million cars—a reasonable annual output for a major EV OEM—each driving 100,000 miles, the overall range increase for that OEM's fleet is five million miles, simply through the use of more efficient HMI code.

Getting More from Less

Without a doubt, electric cars are the future. Three EVs can occupy the same CO_2 footprint as one gasoline-powered economy car. Moreover, that statistic is based on CO_2 figures for U.S. energy production; in countries where electricity generation is less carbon-heavy, the savings are even greater.

Still, charging an EV requires a sizeable amount of electricity. Efficient use of power in every aspect of electric vehicle operation, including software, is important. It reduces CO_2 emissions, lowers charging cost and, perhaps most pragmatically to buyers as EVs proliferate, helps to extend vehicle range. For EVs to change the world we live in, they must go farther—and the software they use must go farther as well.

JEFF STEWART is Altia's Director of Global Sales Engineering. CC HUNG is Altia's Senior Product Marketing Manager.

This article first appeared in [electronics update: Automotive 2021 | volume 5].