Designing Portable Handsets Using CoolRunner-II CPLDs

CPLDs can improve processor-based handsets.

by Mark Ng
Staff Applications Engineer
Xilinx, Inc.
mark.ng@xilinx.com

Portable consumer electronic designs such as cell phone handsets, PDAs, and MP3 players are typically very high-volume products. Because of this, product designers look first to ASIC or ASSP methodologies to pack the greatest functionality into tiny, portable packages.

This solution "hits the mark" for dense functionality and usually has acceptable power consumption. But the consumer world is rapidly changing – features envisioned at one point in time can become obsolete within a matter of months, as competitors react to ever-changing technologies and market dynamics to deliver differentiated solutions.

The term "cutthroat" is frequently used to describe this level of competition. Mistakes are not tolerated, and mistakes are expensive. Choosing the correct ASSP or designing an ASIC correctly every time is nearly impossible. Mitigating these circumstances is crucial to sustaining market share.

Today's designers are now looking beyond the fixed architecture of ASICs and ASSPs to discover the innate design flexibility and time-to-market benefits of programmable logic. Xilinx® CPLDs offer portable device designers a viable alternative to standard cell technology, providing one of the lowest cost, lowest power CPLDs in the industry with CoolRunner™-II devices.

Since 2001, the Xilinx CoolRunner-II CPLD family has provided designers with pricing low enough to beat that of discrete logic devices, allowing designers to easily implement a wide variety of logic functions in a single package. In this article, I'll demonstrate ways to expand beyond the limitations of today's ASIC/ASSP portable handset solutions, with simple, cost-effective, low-power programmable logic using CoolRunner-II CPLDs. As most handsets are OMAP-, XScale-, or i.MX-based designs, I'll describe solutions to several specific problems, with links to application notes that provide in-depth details.

Level Translation

Interfacing two chips of different voltage standards is a common problem. Every type of memory is not made at every voltage standard, and microprocessors are offered at many voltages. Matching standards can be as simple as introducing level translators, but they are expensive and take more area than might be desired. Using a CPLD is a better solution and offers substantially greater flexibility. All CoolRunner-II CPLDs are capable of translating between two voltages, and some can handle as many as four.

CoolRunner-II CPLD I/O banks easily translate between voltages ranging from 1.5 V to 3.6 V in a single chip, as shown in Figure 1. But this totally disregards the programmability of the devices. You get the translation as part of the whole package, which means you get a bundle of logic, flip-flops, power reduction resources, and I/O buffers frequently priced below level translator chips. XAPP785 explains the details on taking advantage of this powerful feature to expand the capabilities of your OMAP, XScale, or i.MX designs.

Pin Expansion

High pin-count ASICs are more expensive than low pin-count ASICs, in general. If your logic needs dictate a low capacity but your I/O requirements dictate a high capacity, you may be paying for logic you will never use to gain the pins. One solution to this is adding a CoolRunner-II CPLD to operate as a "pin expander," as shown in Figure 2.

The basic idea is to identify GPIO pins that typically operate at a slow speed. Then, rather than assign ASIC pins to them, attach CoolRunner-II CPLD pins to the slow-moving GPIO signals, serialize the signals, and import them to the ASIC on fewer net pins. Serializing/deserializing is done through simple, efficient shifting, and can drop the pin counts dramatically on expensive ASICs. Xilinx application note XAPP799 shows how to do this through an I^2C port, but you can use other methods.

As an alternate viewpoint, OMAP, XScale, and i.MX processors provide specific pin mixes to support the applications their vendors deem appropriate. This doesn't mean that you must agree. CoolRunner-II CPLD pin expansion permits you to create your own GPIO pins of assorted voltages and additional capabilities (pulsing, PWM, individually 3-stated).

Pin Swizzling

CPLDs offer the ability to rearrange your pinouts when PCB layout errors occur. This valuable quality is key to keeping you on schedule and within financial and power budgets. Correcting misconnections on a board without having to re-spin the PCB can shave weeks to months off of product schedules.

CoolRunner-II CPLDs are built from powerful logic blocks using programmable logic arrays that can reassign pin logic at will. You will be amazed at how well these devices retain pinouts through multiple edits yet permit re-assigning a design onto different pins as needed. The CoolRunner-II family data sheet explains the architecture and points you to application notes that give all the detail you will need to understand the value of PLAs.

Power Control

Quick power up is one of the strengths of CPLDs. Containing their own configuration cells permits CoolRunner-II CPLDs to power up and direct the activities of other chips as they subsequently arise. This includes some power regulators, which may be sequenced by CoolRunner-II CPLDs, as well as other controlling signals that need to be well defined early in board operation. Xilinx application note XAPP436 describes some of these capabilities.

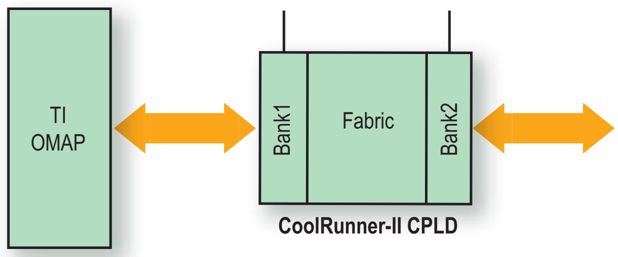

Figure 1 – CoolRunner-II CPLD-level translation of TI OMAP signals

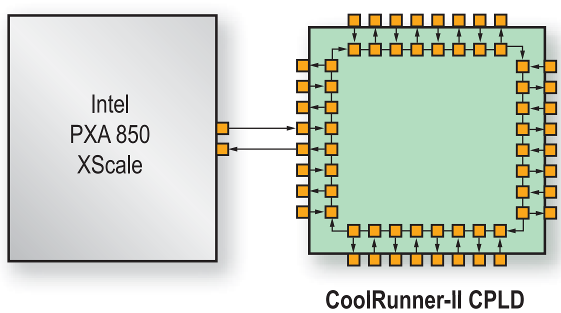

Figure 2 – CoolRunner-II CPLD pin expansion of XScale processor

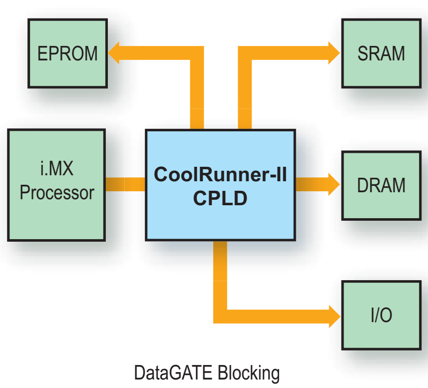

Figure 3 – DataGATE blocking extraneous switching to various devices

Power Reduction

XScale-, OMAP-, and i.MX-based chipsets all include some version of the ARM microprocessor. This is not a surprise. Advanced RISC machines started early with developing low-power methods to operate microprocessors. Subsequently, the licensing vendors have all added their own methods to further reduce processor power. Typical power reduction operations include clock gating, voltage throttling, and on-board memory management to reduce transfers within the device. These are sometimes referred to as run, wait, doze, sleep, and hibernate.

Also, operating systems like Symbian have added "power awareness" to the mix, so that unused resources can be parked in the lowest power mode possible for the current tasks being executed. This all works well and lowers processor power. However, lowering power in the rest of the system exceeds the scope of these methods.

Enter CoolRunner-II CPLDs. CoolRunner-II CPLDs are designed to be inherently low-power parts. That is important, but alone is not enough. CoolRunner-II special features also can be used to lower the power in other devices. Using clock dividers and Xilinx DataGATE technology can reduce power in many (if not all) of the chips on your design, as shown in Figure 3. Blocking power to other chips can also reduce electromagnetic fields being propagated on your board and emanating from your system. This powerful signal blocking technique can pay off in many ways.

Logic Consolidation

Having three two-input AND gates, two three-input OR gates, and a

Schmitt buffer package on your board can burden your bill of materials (BOM), eat away at your power and cost budgets, and lower your reliability. Collecting that stray logic into a consolidated, low-power CoolRunner-II not only solves these problems but stores additional unused logic right there on your board – ready to use with future improvements/edits. WP214 shows what you can expect from collecting logic gates/flip-flops into CoolRunner-II CPLDs. Table 1 summarizes the "burn rate" for logic.

Conclusion

CoolRunner-II CPLDs are quickly becoming the standard for low-power, low-cost, high-volume, portable consumer products. This article has focused on how these powerful products can make life easier when building systems with OMAP, XScale, and i.MX processors, but CoolRunner-II CPLDs work just as well with many other processors to add functionality, save power, and get products to market fast.

Function	Macrocells	P-Terms	Flip-Flops
Shift Register (Simple)	1 per bit	1 per bit	1 per bit
Counter (Simple)	1 per bit	1 per bit	1 per bit
2:1 Mux	1	2	0
4:1 Mux	1	4	0
8:1 Mux	1	8	0
8-bit Loadable Shifter	8	16	8
8-bit Loadable/SL/SR Shifter	8	24	8
8-bit Loadable Counter	8	16	8
8-bit Load/Up/Dn Counter	8	24	8
Full Adder / Bit	2	7	0/1 (optional)
2:4 Decoder	4	4	0
3:8 Decoder	8	8	0
4:16 Decoder	16	16	0
8-bit Equality Comparator	1	16	0
And/Nand Gate (1-40 Inputs)	1	1	0
Or/Nor Gate (1-40 Inputs)	1	11	0
Ex-Or/Ex-Nor (2-3 Inputs)	1	2-3	0
Level Translator (Per Bit)	1	1	0

Table 1 – Macrocell "burn rate" for common TTL functions

TAKE THE NEXT STEP (Digital Edition: www.xcellpublications.com/subscribe/)

- Read the application note, "Level Translation Using Xilinx CoolRunner-II CPLDs."
- Read the application note, "An SMBus/I2C-Compatible Port Expander."
- Read the application note, "Managing Power with CoolRunner-II CPLDs."
- Read the application note, "Using CoolRunner-II Advanced Features" to learn how to do signal blocking.
- Read the white paper, "The Real Value of DataGATE" to learn how much power you can save with DataGATE.