

Customer Specific Standard Products Ease Mobile Device Design



••••• QuickLogic® White Paper

Mobile device designers face a critical question: what functionality should be incorporated in next-generation designs? Answering this question while simultaneously addressing the market needs of rapid development, lowest cost, and long battery life presents a major challenge. The advent of Customer-Specific Standard Products (CSSPs) is helping designers maximize their design options while minimizing cost and effort.

As mobile devices have assumed a central role in consumer electronics, serving both as a personal device and as key core elements in a full-featured home entertainment and communications system, they must provide many connectivity options. Further, the functions of the portable media player, digital camera/camcorder, game system, and enhanced mobile phone with web browsing capabilities have converged into a single mobile device. To enable all these features, design solutions require a variety of interfaces, including high-capacity storage, WLAN, Bluetooth, GPS and others. However, the exact mix of functions that the market will demand is difficult to predict.

Mobile device developers are therefore faced with a major challenge. To succeed in a rapidly changing market their designs must be flexible with a high degree of design reuse or they will take too long in development and miss market opportunities. At the same time, the designs have to achieve the lowest possible cost, so developers cannot hedge their bets by including functions solely for future expansion. Further, the device must offer high performance and yet still enable long battery life.

Therefore, the ideal design approach for mobile devices is one that offers high-integration, design flexibility, ease of implementation and low-power. Unfortunately, conventional design approaches, which include the use of Application-Specific Integrated Circuits (ASICs) and Application-Specific Standard Products (ASSPs), processor-based software implementations and programmable logic, all tend to fall short in one or more of these areas. As a result, designers have had to resort to making trade-offs

Limitations of Current Design Approaches

Application-Specific Integrated Circuits (ASICs)

An ASIC-based design, for instance, can produce components that are relatively inexpensive on a per-unit basis and will yield high-performance and long battery life. They are able to achieve these benefits because ASIC developers can craft a design optimized for performance and battery life while also minimizing die area for lowest cost.

However, there are multiple drawbacks to ASICs that do not make them good choices for many applications.

- **Time to Market** — ASICs take a long time to bring to market – often 12 to 18 months from concept to mass production. In the rapidly-changing mobile device market, only commodity features – ones that customers expect to be in every device – are suitable for ASIC implementation. New and innovative features would take too long to implement in ASIC form, risking late market entry and heavy investment in a function that may not become popular.

- **Cost** — While per-unit mass production costs may be low, ASICs are best suited for production volumes of many millions due to their large initial investment costs. Wafer masks can cost up to \$1,000,000 US for an individual design, depending on the complexity, process node, and other variables. Often, multiple sets of masks need to be ‘cut’ for designs as they move through the evaluation phase, increasing development costs. ASIC developers will require a financial return on these development costs, typically done through amortization of these costs on mass production shipments. If production volume is not high enough, the cost per unit can increase to levels that bill of material (BOM) targets cannot support.

Application-Specific Standard Products (ASSPs)

The ASSP approach tries to improve on ASIC development by offering components that are mostly pre-designed, awaiting only software configuration during system boot-up to complete the implementation. This approach is relatively inflexible, however, with limited opportunities for customization. It relies solely on the ASSP vendor to have pre-defined exactly the right mix of capabilities in the silicon. Otherwise, the component will have a silicon area dedicated to functions that are not being used, inflating production costs. Also, while shorter than full ASIC implementations, ASSP design times are still long enough to risk missing market opportunities.

Software Implementations

Processor-based software implementations offer a high degree of design flexibility by reprogramming the processor to add new functions. They also provide an opportunity for design reuse to speed development of derivative and next-generation devices. While the processor-based approach may have been optimal for early mobile devices, today’s designs are so function rich that a pure software-based implementation requires substantial processing capability. That, in turn, demands high power levels that can severely reduce battery operating time. Building peripherals into the processor helps address these concerns, but introduces the same costs and inflexibility issues that ASSPs face.

Programmable Logic and Complex Programmable Logic Devices (CPLDs)

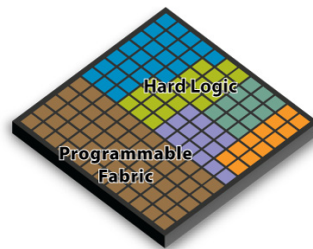
Like processors, programmable logic also offers flexibility and design reuse benefits. By itself, however, programmable logic cannot address all mobile device requirements. Primarily, this is due to the fact that SRAM-based programmable logic technologies are notorious for their power demands, high cost and volatility. Typically, the only programmable logic used in a mobile device is in the form of a Complex Programmable Logic Device (CPLD) – a true misnomer since CPLDs can really only handle glue logic and GPIO functionality.

Typically, a mobile device design will use a blend of these techniques. Software handles some functions, ASSPs and other standard product peripherals handle I/O interfaces, and occasionally a CPLD will glue them together. Unfortunately, the blended approach is not particularly cost competitive because it requires too many components and results in a relatively large and inefficient design.

The CSSP Concept

The design alternative to the traditional methods previously described is the CSSP pioneered by QuickLogic. This design solution works as a companion device to the host processor and offers developers a unique combination of flexibility, integration, and silicon efficiency while keeping power demands low. Like an ASSP, CSSPs integrate the *application-specific* functions that are most common to the mobile market in hard logic. At the same time, CSSPs also integrate a programmable fabric as shown in **Figure 1**, which allows additional *customer-specific* functionality to be integrated into the CSSP quickly and efficiently, enabling differentiation at the silicon level.

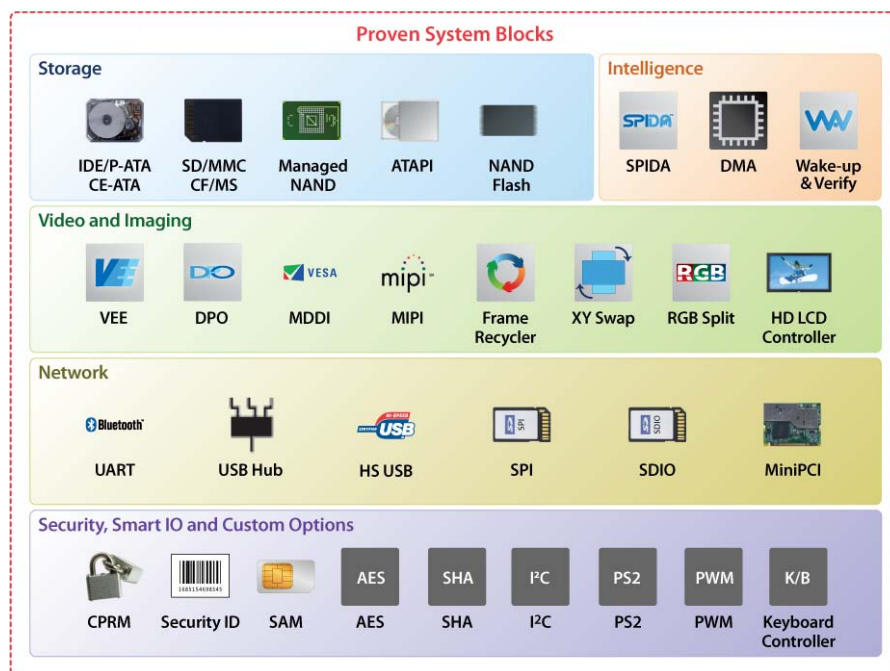
Figure 1: CSSPs integrate common mobile market application-specific functions in the hard logic and customer-specific functionality in the programmable fabric.



The interface to the application processor typifies how the method works. Rather than implementing the address and data lines, bit order, interrupt signals, and such for a specific processor, the CSSP processor interface offers generic signal lines. These lines do not remain generic, however. When the developer chooses the specific application processor to be used, the signal lines can then be configured with the proper interpretations, bit order, edge sensitivity and other details to become an exact match to the chosen processor. A proven library of such implementations ensures that the interface will function correctly and with optimized power and performance.

Because no one can accurately predict the exact mix of functions that a next-generation mobile device developer will require, the CSSP also offers a low-power, non-volatile programmable fabric to augment the pre-defined standard interfaces. Developers can select from a library of functions, called Proven System Blocks (PSBs) as shown in **Figure 2**, for implementation in this fabric, allowing them to define exactly the combination of functions they require. They can also achieve product differentiation by implementing custom designs in the fabric to provide new and innovative functions. This combination of pre-defined blocks and programmable fabric provides developers the exact mix they require, with an overall silicon efficiency better than that of programmable logic devices. A single CSSP can replace as many as five separate devices that might be needed for a blended design.

Figure 2: Proven System Block options from QuickLogic.



The CSSP addresses the full range of needs for mobile device designers.

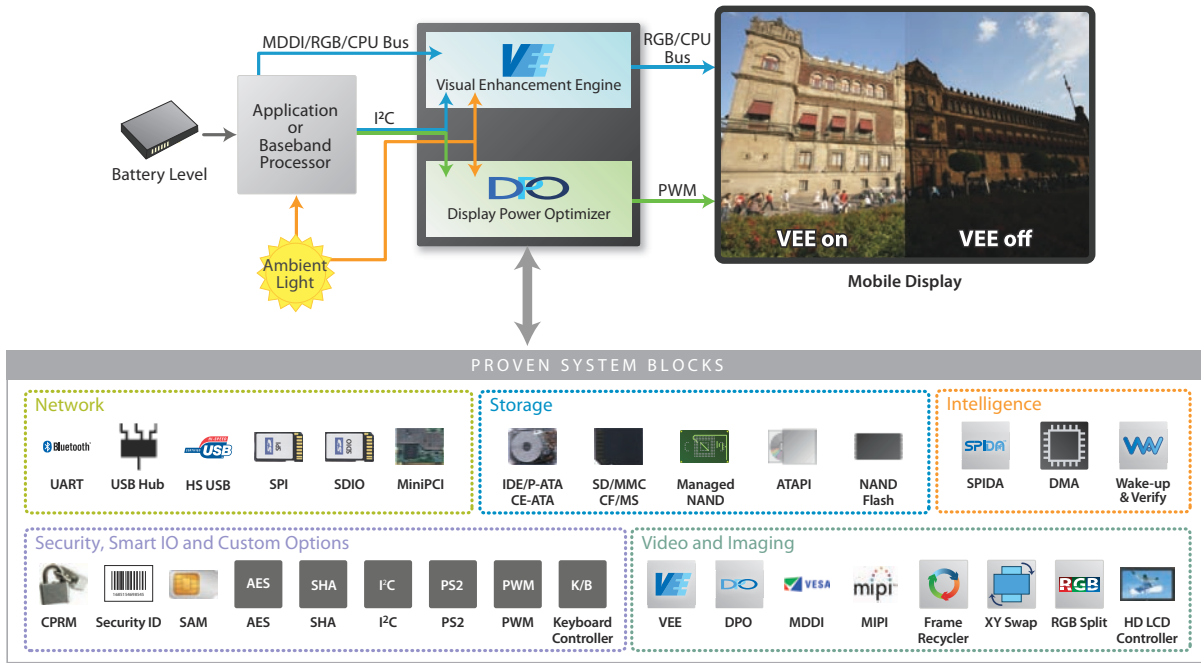
- Silicon efficiency — CSSP silicon efficiency helps keep production costs low.
- Key pre-configured functions — Key functions and technologies are pre-configured on the CSSP. These functions are optimized to offer high performance, and the underlying low-power silicon technology ensures long battery life.
- Design time — The development time needed to craft a customer-specific component is short because the design is based on a standard product that only awaits factory configuration to finish its implementation. Similarly, the development effort for creating one design can be re-used and quickly adapted to create derivative products or to address changes in market requirements simply by making changes to the programmable fabric of the CSSP. Design times on CSSPs have been as short as four weeks from concept to prototype silicon, with average times from concept to mass production of less than five months.

The ArcticLink[®] II Solution Platform for CSSPs

QuickLogic's ArcticLink II solution platform typifies the CSSP approach. The architecture of the ArcticLink II solution platform, shown in **Figure 3** contains several different blocks. The hard logic portion of the device contains several blocks. The first is the Visual Enhancement Engine (VEE). VEE technology greatly enhances the viewability of displays under challenging conditions such as bright ambient lighting while dynamically optimizing video characteristics on a pixel-by-pixel, frame-by-frame basis to produce a natural viewing experience for the user. The second is the Display Power Optimizer (DPO) which enables consumer device developers to reduce system battery consumption by as much as 25% through lowering display backlight or power intensity without compromising the viewing experience. Also included in the ArcticLink II hard logic is either an MDDI type 2 client (60 frames per second) or RGB interface to the CPU, as well as an I²C client and a Pulse Width Management (PWM) function.

The processor interface block can be configured to handle a wide range of popular application and baseband processors for mobile designs. Library functions for Freescale™, Intel®, Marvell®, Qualcomm®, Samsung™, Texas Instruments™ and other processors are available. Developers can also choose to create interfaces for other processors if desired. Finally, a programmable fabric within the ArcticLink II solution platform is available for developers to implement additional, customer-specific functions.

Figure 3: By blending configurable interface blocks and a programmable fabric, the ArcticLink II solution platform CSSP offers developers a wide range of connectivity options.



Design Use Case Examples

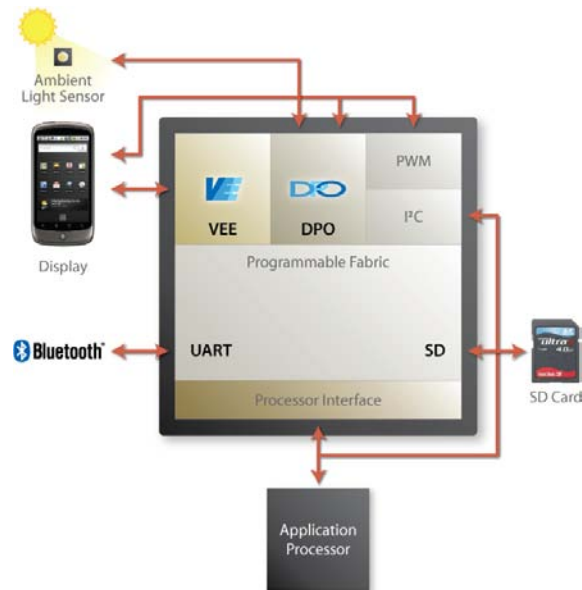
The following design examples use the ArcticLink and ArcticLink II solution platforms to demonstrate the versatility of the CSSP approach.

Smartphones

The design shown in **Figure 4** for instance, is implemented in a smartphone for “road warriors” like Brian. As a heavy user of his handset, Brian needs a phone with long battery life and a superior user experience. Hard-logic VEE functionality allows Brian to view video and display content as he travels by train to his next customer, in and out of tunnels, through bright sunlight and clouds, without having to manually adjust the backlight or relocate to a different viewing environment. All the while, DPO is consistently working to conserve display power consumption, allowing Brian to continue working (or, watching the latest episode of American Idol) on his handset for 25% or longer than without DPO. In the programmable fabric, the CSSP features a Bluetooth 2.0 UART, allowing Brian to drive his car and speak on the phone using the hands-free connection, saving him the valuable time of having to stop his car to make phone calls (or, the cost of a traffic ticket for driving and talking on a phone without a hands-free device). Finally, an SD memory interface is built into the

programmable CSSP fabric, allowing connection to a high-capacity SD memory card. This allows Brian to store large presentations and documents on his phone and, though he may not readily admit it, a copy (legal, of course) of ABBA's greatest hits.

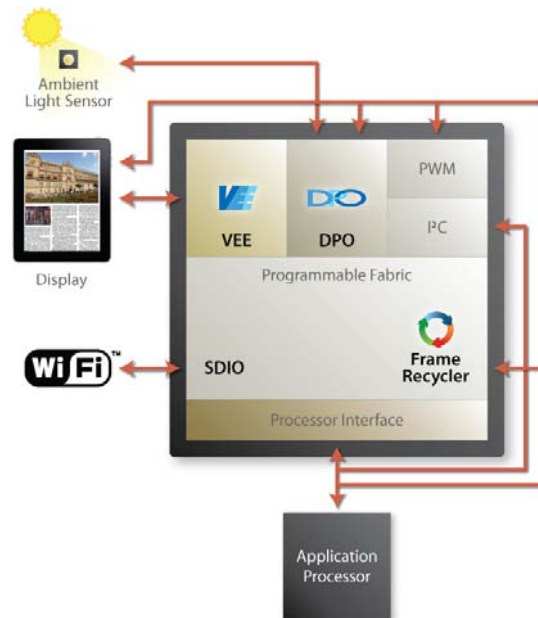
Figure 4: By configuring its blocks for VEE, DPO, Bluetooth, and an SD memory controller, the ArcticLink II solution platform becomes an integrated processor companion for a smartphone.



Tablets/Smartbooks

Mehul has recently purchased his first tablet computer, and he hopes to use it on the car trip he has planned for his family from San Francisco to San Diego. Since the trip is almost 8 hours, Mehul hopes that the battery of his new electronic gadget will last long enough to keep him and his family entertained on the long trip. The design shown in **Figure 5** is implemented with Mehul's goals in mind. As the trip begins, Mehul's son begins watching a video on the tablet. The hard-logic integrated VEE functionality allows a superior viewing experience as they travel down the California coast, whether in and out of tunnels on Route 1, through the sun of Monterey, or the fog of Santa Barbara. Not once is Mehul forced to adjust the backlight intensity or contrast of the display to be able to comfortably view the display. At the same time, DPO is consistently working in the background to conserve display power consumption, allowing Mehul's son to continue viewing video without Mehul having to worry about the tablet running out of power. After his son falls asleep, Mehul takes the tablet and uses it as an e-reader, starting first with his favorite magazine, *LPGA Weekly*. The implementation of the Frame Recycler PSB in the programmable fabric portion of the CSSP allows the display portion of the CPU to shut down, conserving significant system power. Finally, the SDIO interface in the CSSP fabric provides for a wireless connection, which is very timely, given that a check of traffic conditions over the wireless connection during their lunch stop shows a massive traffic jam just north of Los Angeles.

Figure 5: By configuring its blocks for VEE, DPO, Frame Recycler, and an SDIO wireless interface, the ArcticLink II solution platform becomes an integrated processor companion for a tablet computer.

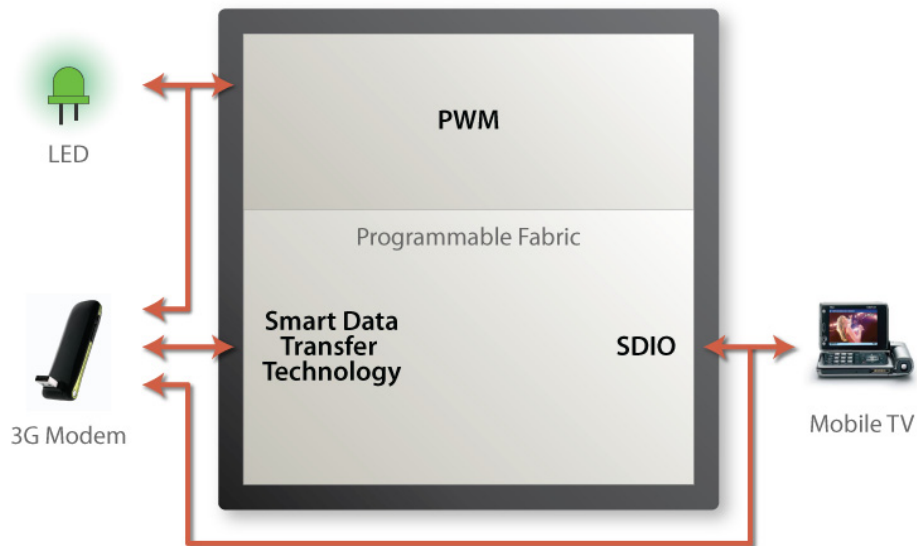


Datacards

Like a lot of people, Andy is extremely impatient for data; also like many others, Andy is habitually late. The example architecture shown in **Figure 6** is well suited for him. On a certain Friday, Andy wakes late to realize he missed his alarm and has only a few minutes to make his first meeting of the day. Andy rushes out of the house and jumps into a taxi cab on the way to his customer. Frantic, he calls his co-workers for help with a presentation he was supposed to prepare that morning. Luckily, his co-workers anticipated this, so they prepared and e-mailed the presentation to him the evening before. Relieved, he begins to download the presentation to his computer using a 3G datacard. As downloading begins, the PWM function of the QuickLogic CSSP ArcticLink solution platform controls the datacard LEDs, providing Andy a visual way of confirming operation and download. He is able to download the entire 20 MB presentation using the QuickLogic Smart Data Transfer (SDT) technology in half the time it would have taken without it – giving him time to review his co-workers' superior presentation workmanship in the last few minutes of the taxi ride. The presentation goes smashingly well.

In the less-harried taxi ride back to his office, Andy enables the mobile TV function of the 3G datacard, which functions through the CSSP SDIO port, allowing him to watch the highlights from the previous night's football games. Seeing that his favorite team won and their hated rivals lost, he decides to stop in the local pub for a quick celebratory 'lunch' with friends.

Figure 6: By configuring the ArcticLink solution platforms programmable fabric with the QuickLogic SDT technology, Andy is able to download items much quicker



Summary

These three diverse applications can all be implemented with the same base standard products, the ArcticLink and ArcticLink II solution platforms, differing only in their final configuration. Because this configuration step occurs after final silicon fabrication and test, the three CSSPs draw on the same base silicon, allowing them to share the cost benefits of volume production. Thus, the CSSP approach yields a new combination of design flexibility, rapid development, and low cost that will help ease the burden on mobile device designers.

Contact Information

Phone: (408) 990-4000 (US)
(647) 367-1014 (Canada)
+(44) 1932-21-3160 (Europe)
+(886) 2-2345-5600 (Taiwan)
+(86) 21-5116-0532 (China)

E-mail: info@quicklogic.com

Sales: America-sales@quicklogic.com
Europe-sales@quicklogic.com
Asia-sales@quicklogic.com
Japan-sales@quicklogic.com

Support: www.quicklogic.com/support

Internet: www.quicklogic.com

Revision History

Revision	Date	Originator and Comments
A	June 2007	Howard Li and Kathleen Murchek
B	July 2007	Howard Li and Kathleen Murchek Updated graphics for consistency.
C	November 2008	Kathleen Murchek Updated banner.
D	January 2009	Kathleen Murchek Updated contact information, copyright, and added disclaimer.
E	April 2009	Kathleen Murchek Updated trademark information.
F	August 2010	Paul Karazuba and Kathleen Bylsma Major rewrite, new graphics and updated contact information.

Notice of Disclaimer

QuickLogic is providing this design, product or intellectual property "as is." By providing the design, product or intellectual property as one possible implementation of your desired system-level feature, application, or standard, QuickLogic makes no representation that this implementation is free from any claims of infringement and any implied warranties of merchantability or fitness for a particular purpose. You are responsible for obtaining any rights you may require for your system implementation. QuickLogic shall not be liable for any damages arising out of or in connection with the use of the design, product or intellectual property including liability for lost profit, business interruption, or any other damages whatsoever. QuickLogic products are not designed for use in life-support equipment or applications that would cause a life-threatening situation if any such products failed. Do not use QuickLogic products in these types of equipment or applications.

QuickLogic does not assume any liability for errors which may appear in this document. However, QuickLogic attempts to notify customers of such errors. QuickLogic retains the right to make changes to either the documentation, specification, or product without notice. Verify with QuickLogic that you have the latest specifications before finalizing a product design.

Copyright and Trademark Information

Copyright © 2010 QuickLogic Corporation. All Rights Reserved.

The information contained in this document is protected by copyright. All rights are reserved by QuickLogic Corporation. QuickLogic Corporation reserves the right to modify this document without any obligation to notify any person or entity of such revision. Copying, duplicating, selling, or otherwise distributing any part of this product without the prior written consent of an authorized representative of QuickLogic is prohibited.

QuickLogic and ArcticLink are registered trademarks, and the QuickLogic logo is a trademark of QuickLogic. Other trademarks are the property of their respective companies.