Please note that Cypress is an Infineon Technologies Company.
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

THIS SPEC IS OBSOLETE

Spec No: 001-46712

Spec Title: INTERFACING TO WEST BRIDGE(R) ASTORIA(TM) PSEUDO-NAND PROCESSOR PORT - AN46712

Sunset Owner: Dhanraj Rajput (DBIR)

Replaced by: None

AN46712

Interfacing to the West Bridge® Astoria™ Pseudo-NAND Processor Port

Author: Bahram Raad
Associated Project: No
Associated Part Family: CYWB0224ABS/M
Software Version: Astoria™ SDK Version 1.2.1
Related Application Notes: None

West Bridge® Astoria™ has a robust feature that enables interfacing with a system processor using a standard NAND interface and common NAND Flash commands. This feature allows Astoria to emulate a NAND Flash device, which enables a system processor to boot from Astoria. This application note describes how to interface a system processor to the Pseudo-NAND interface of Astoria.

Contents

Introduction

West Bridge® Astoria™ supports multiple Processor-Port (P-port) interfaces that enable Astoria to connect to commonly used system processors. One of the unique interfaces available in Astoria is the Pseudo-NAND (PNAND) P-port interface. The PNAND interface allows a system processor to access Astoria using a NAND Flash controller. In some systems, it is advantageous to allow the system processor to boot directly from NAND Flash. Astoria's PNAND boot feature enables a system processor to communicate with Astoria using standard NAND commands, and boot directly from a NAND Flash connected to Astoria. The PNAND boot feature enables robust system design by consolidating boot and mass storage data in a single SLC NAND device. This feature reduces BOM cost by removing the need for two NAND devices on the board: one for booting the processor and other for mass storage. This application note describes how a system processor can interface with Astoria through the PNAND interface, and explains the considerations involved.

West Bridge Astoria Overview

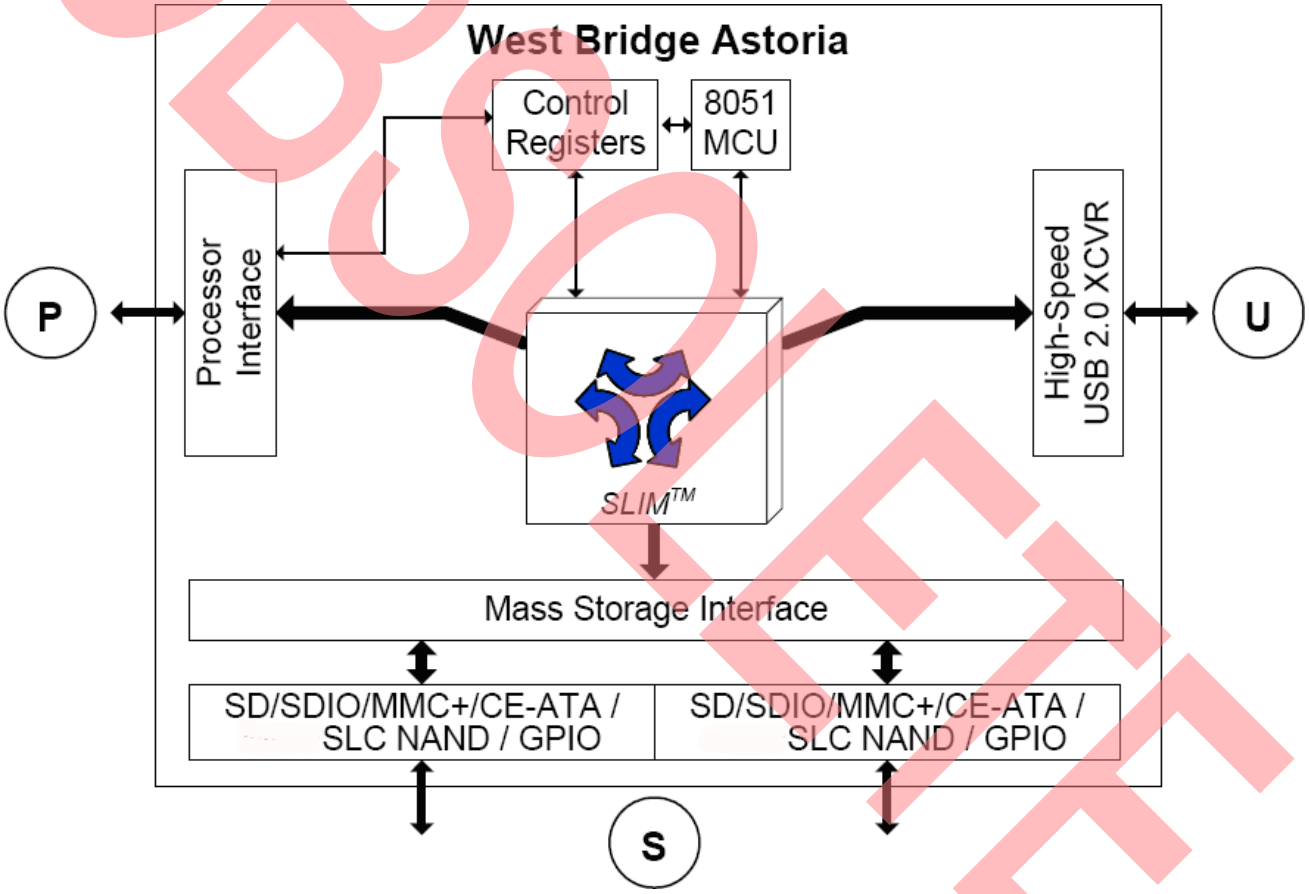
Astoria is a flexible bridge device that uses robust SLIM® architecture, which allows simultaneous and independent data paths between the processor and USB, USB and mass storage, and processor and mass storage.

The Processor-port (P-port) of Astoria can operate in one of the multiple selectable interface types such as CRAM, SRAM, Address-Data Multiplexed RAM, SPI, or PNAND.

The USB-port (U-port) is a High-Speed (HS) USB peripheral that can transfer independently to and from the processor and mass-storage media with optimum performance.

The mass storage-port (S-port) can be configured to simultaneously interface with multiple mass-storage devices such as 8-bit or 16-bit SLC NAND, Managed NAND, SD/MMC/SDIO, and CE-ATA devices.

Figure 1. West Bridge Astoria Block Diagram



For a full description of Astoria's features, see the West Bridge Astoria data sheet.

PNAND Overview

Astoria's processor port NAND interface is named Pseudo-NAND (PNAND) interface, because although its functions are similar to a NAND device, there are some key differences. There are two modes of operation in the PNAND interface:

- Logical NAND Access
- Non-Logical NAND Access

Logical NAND Access (LNA) refers to the mode of operation where Astoria emulates a NAND Flash device. This mode is designed for systems that require booting of the system processor from a NAND Flash device. In this type of application, the system processor can communicate to Astoria using common NAND commands to boot from a NAND Flash connected to Astoria's S-port. In this mode of operation, Astoria mimics a real NAND device and allows the system processor to use its internal boot-ROM to boot from Astoria, as it would boot from a NAND Flash.

In non-LNA mode of operation, the system processor interfaces with Astoria using standard NAND interface, but does not use standard NAND commands. In non-LNA mode, Astoria responds to a subset of NAND commands. In this mode of operation, the system processor uses a set of APIs provided by Cypress to communicate through its NAND controller to Astoria. The non-LNA mode of operation can be used independent of the PNAND LNA boot.

When Astoria is used in PNAND LNA mode, its available features are limited to transfers between P-port and U-port, and NAND Flash access on the S-port. To take advantage of Astoria's features, the system must switch from LNA to non-LNA mode of operation by setting a register after booting is complete. The register setting is handled through the APIs. Although P⇔U access is available in LNA mode, the Astoria APIs must be used to access this feature.

The PNAND interface of Astoria can be configured to support one of the following modes:

- 8-bit Small Block Device (SBD)
- 8-bit Large Block Device (LBD)
- 16-bit Small Block Device
- 16-bit Large Block Device

In SBD mode, there is one address cycle for column address and three address cycles for row address whereas in LBD mode, there are two address cycles for column address and three address cycles for row address. Moreover, in SBD mode, the acceptable commands are 00h, 01h, or 50h whereas in LBD mode, the only command is 00h.

SBD or LBD mode is selected through external pin settings as shown in Table 1. The 8-bit versus 16-bit mode of operation is selected by the firmware through internal register settings.

Physical Interface to PNAND P-Port

Figure 2 shows the physical connection between a system processor and Astoria's PNAND interface. Note that the S-Port may have other devices such as SD/MMC/CE-ATA or multiple NAND devices connected to it. The I2C EEPROM is only required for Astoria's self boot option.

Figure 2. Example of a Connection between a Processor's NAND Controller and Astoria

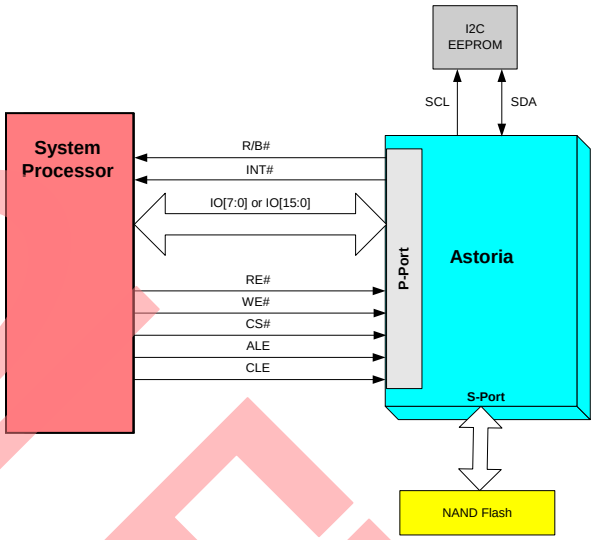


Table 1 describes the signals in the PNAND interface of Astoria.

Table 1. Astoria PNAND Interface Signals

P-port signal of Astoria	I/O	PNAND signal	Function	Comments
CE#	I	CS#	Chip Select	
A[0]	I	CLE	Command Latch Enable	
A[1]	O	R/B#	Ready/Busy#	This is not an open-drain signal
A[3:2]	I	A[3:2]	PNAND mode select	A[3:2] = 2`b00
A[4]	I	WP#	Write Protect	Optional
A[5]	O	SCL	I2C serial clock	Used for EEPROM boot. Must be pulled up.
A[6]	I/O	SDA	I2C serial data	Used for EEPROM boot. Must be pulled up.
A[7]	I	SBDnLBD	SDB or LBD select pin	A[7] = 1 => SBD A[7] = 0 => LBD
DQ[15:0]	I/O	IO[15:0]	Data bus	May be used as 8 or 16 bits
ADV#	I	ALE	Address Latch Enable	
OE#	I	RE#	Read Enable	
WE#	I	WE#	Write Enable	
INT#	O	INT#	Interrupt Request	Optional. Useful in non-LNA
DRQ#	O	DRQ#	DMA Request	Optional in non-LNA mode
TEST[2:0]	I	TEST[2:0]	Mode configuration	Must configure to PNAND mode

There are different modes of operation within PNAND that affect the pin connectivity listed in Table 1. This is explained in successive sections of this application note.

To configure Astoria for PNAND SBD or LBD mode of operation, the external pins must be configured as shown in the following table.

Table 2. PNAND Mode Configuration Pin Setting

Mode	TEST[2]	TEST[1]	TEST[0]	A[7]	A[3]	A[2]
PNAND SBD	0	1	0	1	0	0
PNAND LBD	0	1	0	0	0	0

The 8-bit versus 16-bit mode of operation is selected internally through Astoria firmware and can be reconfigured through Astoria APIs.

PNAND LNA Mode

The PNAND LNA mode is typically used when a system processor wants to boot from a NAND Flash storage device. In this mode, Astoria emulates a NAND device and responds to common NAND commands. In this method, the processor can boot directly from a NAND Flash connected to Astoria's S-port without having to make changes to its boot-ROM.

Consider the following points when using PNAND LNA boot:

- Booting Astoria
- Differences between real NAND and Astoria's NAND emulation mode
- Commands supported by Astoria's PNAND interface

Booting Astoria

Astoria has a built in 8051 MCU that handles several tasks internally, including NAND emulation. The internal instruction memory is RAM based. To use the PNAND boot option, Astoria must load its instruction RAM first. There are two options available for booting Astoria in this mode:

- External I2C EEPROM
- Internal boot-ROM

Astoria can boot from an external I2C EEPROM. With this option, Astoria initially boots itself from an EEPROM and then sets R/B# signal to Ready to signal readiness for processor boot.

To configure Astoria for booting from an external I2C EEPROM, the external pins must be configured to select one of the PNAND modes as shown in Table 2 on page 3.

When Astoria is configured for PNAND mode as shown in Table 2, upon reset it checks for a valid EEPROM device connected to its I2C port. If the I2C EEPROM ACKs the expected I2C address, the boot-ROM proceeds to check for an expected pattern at the initial address range of the EEPROM. The pattern at the header of the EEPROM boot code is a Cypress proprietary standard that is provided to a customer upon request. The EEPROM's configurable address pins must be set according to the following table.

Table 3. I2C Boot EEPROM Address Configuration

Bytes	EEPROM	A2	A1	A0
4K	24LC32	0	0	1
8K	24LC64	0	0	1
16K	24LC128	0	0	1
32K	24LC256	0	0	1

Another option for booting Astoria is to use the internal boot-ROM. Note that Astoria silicon is not normally available with a built in Internal boot-ROM. There is an option to create an internal boot-ROM based on customer request. This option is not readily available in Astoria and must be enabled based on customer engagement. This option enables the stand alone operation of Astoria, without the need for external instruction RAM loading.

Differences between NAND Flash and Astoria PNAND

There are some differences between NAND Flash and Astoria's NAND emulation interface. The following table describes these differences.

Table 4. Differences between NAND Flash and Astoria PNAND LNA Interface

Feature	NAND Flash	PNAND
R/B signal	Open drain signal allowing multiple NAND device connectivity.	Driven signal.
tR, tPROG, tBERS, tADL	These timing parameters vary across different NAND parts.	Typically larger than the NAND Flash because of added processing in Astoria. For exact timing parameters, refer to data sheet.
Command set	Typically support a basic command set plus additional manufacturer-specific commands.	Supports common NAND command set as noted in Tables 5 and 6.
Random data input (CASDI)	LBD devices support random data input (CASDI) to Flash array.	Random data input to NAND Flash is not supported. Random data input may be used for register writes in LBD mode.

Various NAND Flash devices may use different NAND commands.

The PNAND interface uses a subset of the basic NAND commands. Table 5 and Table 6 on page 5 outline the commands supported by the NAND interface.

Table 5. Supported PNAND commands in LBD mode

Command	Command		Supported Mode		Notes
	1st Cycle	2nd Cycle	LNA	non-LNA	
Page Read	00	30	Y	Y	Used for both Flash access and register Read operation
Read ID	90	NA	Y	N	
Reset	FF	NA	Y	N	
Page Program	80	10	Y	Y	Used for both Flash programming and register Write operation.
Multi-Plane Page Program	80--80	11	Y	N	Command is treated exactly the same as Page Program with tPROG = tDBSY for each plane.
Block Erase	60	D0	Y	N	
Multi-Plane Block Erase	60--60	D0	Y	N	Requires 450 ns delay between 1st and 2nd Plane address sent.
Random Data Input (CASDI)	85	E0	Y	Y	This command is only supported for register writes.
Random Data Output (CASDO)	5	E0	Y	Y	In non-LNA mode, it is only supported for register access.
Read Status	70	NA	Y	N	
Read Status Multi-Plane	71	NA	Y	N	

Table 6. Supported PNAND commands in SBD mode

Command	Command		Supported Mode		Notes
	1st Cycle	2nd Cycle	LNA	non-LNA	
Page Read (CMD00)	00	30	Y	Y	Used for both Flash access and register Read operation. Sequential Read is not supported.
Page Read2 (CMD01)	01	30	Y	N	Used for Flash access only. Sequential Read is not supported.
Page Read3 (CMD50)	50	30	Y	N	Used for Flash access only. Sequential Read is not supported.
Read ID	90	NA	Y	N	
Reset	FF	NA	Y	N	
Page Program	80	10	Y	Y	Used for both Flash programming and register Write operation
Multi-Plane Page Program	80--80	11	Y	N	Command is treated exactly the same as page program with tPROG = tDBSY for each plane.
Block Erase	60	D0	Y	N	
Multi-Plane Block Erase	60--60	D0	Y	N	Requires 450 ns delay between 1st and 2nd Plane address sent.
Read Status	70	NA	Y	N	
Read Status Multi-Plane	71	NA	Y	N	

As reference for the NAND commands that are supported by Astoria, refer to the data sheet for the following NAND devices:

- K9W8G08U1M Large block NAND Flash Device
- K9K1G08R0B Small block NAND Flash Device

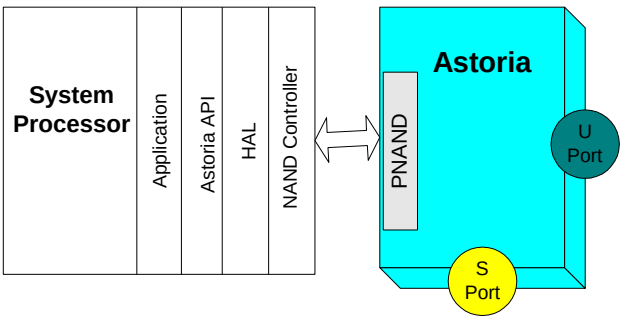
PNAND non-LNA Mode

PNAND non-LNA mode of operation provides a NAND type interface to Astoria. As noted in previous sections, SBD or LBD, and 8-bit or 16-bit configurations may be used. Standard NAND controllers on system processors can use this interface to access all features of Astoria.

The major difference between LNA and non-LNA modes of operation is that in LNA mode, Astoria emulates a NAND device. This removes the need for having APIs on the processor side for communicating to Astoria. In contrast, in non-LNA mode a set of proprietary commands are used to communicate to the PNAND interface. These proprietary commands are facilitated through a set of APIs provided by Cypress. For example, to read one page of data from a NAND Flash device connected to the S-Port of Astoria in non-LNA mode, the system processor must communicate to Astoria through the APIs, instead of sending a NAND Page Read command.

Figure 3 illustrates a high level view of PNAND non-LNA connectivity between the system processor and Astoria. Astoria software APIs are required for communication through the processor NAND controller to Astoria's PNAND non-LNA interface.

Figure 3. High Level View of Processor Connectivity to Astoria in Non-LNA Mode



The NAND commands used to access Astoria in non-LNA mode are listed in Table 5 and Table 6 on page 5.

In PNAND non-LNA mode, it is recommended to connect the INT# pin of Astoria to the processor for detecting various interrupts. In this mode, the R/B# signal can be used to detect the state of the P⇔U and P⇔S endpoints.

In some systems, booting from PNAND (LNA mode) may not be required but it may be desired to use PNAND non-LNA as the interface to Astoria. In this case, the following options can be used to download firmware to Astoria:

- **Load firmware from processor.** Astoria APIs support a method of loading firmware through P-Port to the internal instruction RAM of Astoria.
- **Load firmware from I2C EEPROM**. This is the same method of firmware loading described above for PNAND LNA mode

Loading firmware from processor is desirable in most cases, because it eliminates the need for an additional EEPROM component.

Astoria SDK

Astoria Software Development Kit (SDK) is provided with a set of components to enable quick and easy system integration into customer system. The SDK provides the following components:

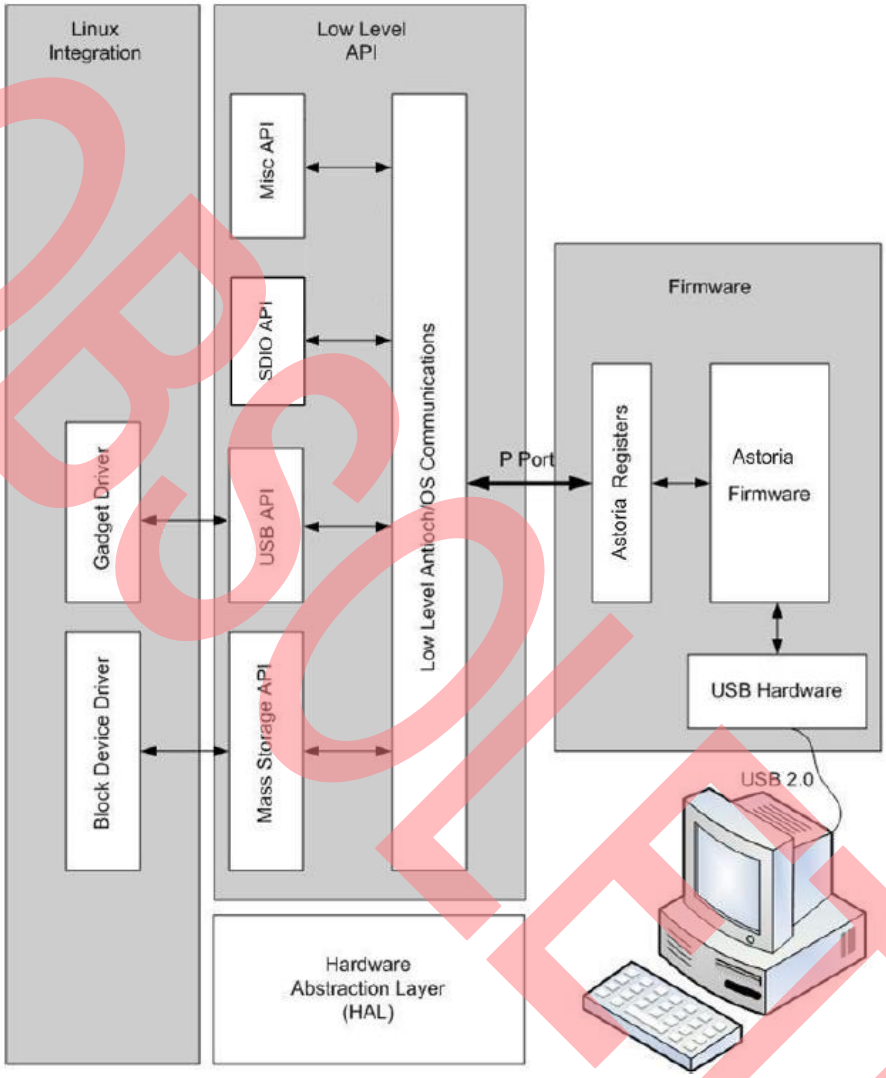
- Firmware images: Astoria firmware in binary format for various application scenarios.
- Astoria API: API commands for interfacing system processor applications to Astoria.
- Documentation

Astoria APIs provide a flexible set of functions to access all Astoria features. These APIs are required in non-LNA mode of operation. APIs are written in C and designed for easy portability to various architectures.

Figure 4 illustrates a typical Linux-based system using Astoria APIs for communicating with Astoria.

For more details on Astoria SDK, refer to "West Bridge® Astoria™ SDK User Guide".

Figure 4. High-Level View of Astoria System Software



Summary

Astoria's PNAND interface provides an option of interfacing to systems that require access to USB and various mass storage media, through a NAND physical interface. By supporting multiple modes of operation for LBD, SBD, 8-bit, or 16-bit, the PNAND interface is flexible in compatibility with NAND controllers on different system processors.

The PNAND emulation mode provides a flexible method of booting a system processor from a NAND device attached to Astoria's storage port, while simultaneously adding High-Speed USB and mass-storage capability to the processor. This feature reduces system complexity and cost by consolidating the processor boot memory and system mass-storage memory into a single SLC NAND device.

Related Documents

West Bridge Astoria Advance Datasheet

CYWB022XX Family Datasheet

Integrating West Bridge Astoria with Android

Document History

Document Title: Interfacing to the West Bridge® Astoria™ Pseudo-NAND Processor Port

Document Number: 001-46712

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	2511898	JASM	07/03/2008	New application note.
*A	3183319	ANOP	02/27/2011	Updated for Astoria SDK version 1.2.1, updated trademark usage.
*B	3592072	AASI	04/23/2012	- Figure 1 updated to exclude MLC. - Table 4, row 2, column 2, the statement 'These timing parameters vary across different NAND parts and technologies such as SLC/MLC' changed to 'These timing parameters vary across different NAND parts' - Added the following paragraph to the 'PNAND Overview' section - 'In SBD mode, there is one address cycle for column address and three address cycles for row address whereas in LBD mode, there are two address cycles for column address and three address cycles for row address. Moreover, in SBD mode, the acceptable commands are 00h, 01h or 50h whereas in LBD mode, the only command is 00h.'. - Added the following links under “Related Documents” section: o CYWB022XX Family Datasheet (http://www.cypress.com/?rID=60321) o 001-59040 - Integrating West Bridge Astoria with Android (http://www.cypress.com/?rID=40254)
*C	4108854	RSKV	08/30/2013	Obsolete application note

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Automotive	cypress.com/go/automotive
Clocks & Buffers	cypress.com/go/clocks
Interface	cypress.com/go/interface
Lighting & Power Control	cypress.com/go/powerpsoc cypress.com/go/plc
Memory	cypress.com/go/memory
Optical Navigation Sensors	cypress.com/go/ons
PSoC	cypress.com/go/psoc
Touch Sensing	cypress.com/go/touch
USB Controllers	cypress.com/go/usb
Wireless/RF	cypress.com/go/wireless

PSoC® Solutions

psoc.cypress.com/solutions

PSoC 1 | PSoC 3 | PSoC 5

Cypress Developer Community

Community | Forums | Blogs | Video | Training

Technical Support

cypress.com/go/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone : 408-943-2600
Fax : 408-943-4730
Website : www.cypress.com