MICRONAS

Hardware Documentation

Preliminary Data Sheet

Cypher™ ESN7108A

VIRTUAL DOLBY SURROUND

ViVA HD3D Sound BBE®

MICRONAS BASS

MICRONAS AROUND

DOLBY DIGITAL

HD ready

MPEG

SPATIALIZER® N-2-2 ULTRA™

Edition September 2007
6251-695-3PD

Copyright, Warranty, and Limitation of Liability

The information and data contained in this document are believed to be accurate and reliable. The software and proprietary information contained therein may be protected by copyright, patent, trademark and/or other intellectual property rights of Micronas. All rights not expressly granted remain reserved by Micronas.

Micronas assumes no liability for errors and gives no warranty representation or guarantee regarding the suitability of its products for any particular purpose due to these specifications.

By this publication, Micronas does not assume responsibility for patent infringements or other rights of third parties which may result from its use. Commercial conditions, product availability and delivery are exclusively subject to the respective order confirmation.

Any information and data which may be provided in the document can and do vary in different applications, and actual performance may vary over time.

All operating parameters must be validated for each customer application by customers' technical experts. Any new issue of this document invalidates previous issues. Micronas reserves the right to review this document and to make changes to the document's content at any time without obligation to notify any person or entity of such revision or changes. For further advice please contact us directly.

Do not use our products in life-supporting systems, aviation and aerospace applications! Unless explicitly agreed to otherwise in writing between the parties, Micronas' products are not designed, intended or authorized for use as components in systems intended for surgical implants into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the product could create a situation where personal injury or death could occur.

Micronas Trademarks

– Cypher

Third-Party Trademarks

All other brand and product names or company names may be trademarks of their respective companies.

Contents

Contents, continued

Contents, continued

Page Section Title

1. Introduction

The Cypher ESN7108A is a highly integrated video/audio encoder offering high-quality real time MPEG-4/2/1/H.263/MJPEG video and Dolby Digital® (AC-3/MPEG) audio streaming and capture from full D1 CCIR-656/601 YUV video sources. The integrated I/O interfaces of the ESN7108A SoC make it an ideal choice for embedded applications such as consumer electronics, PC/MAC, and video surveillance.

Figure 1–1 shows a block diagram of the Cypher ESN7108A MPEG Encoder. Each of the blocks is described in the following subsections.

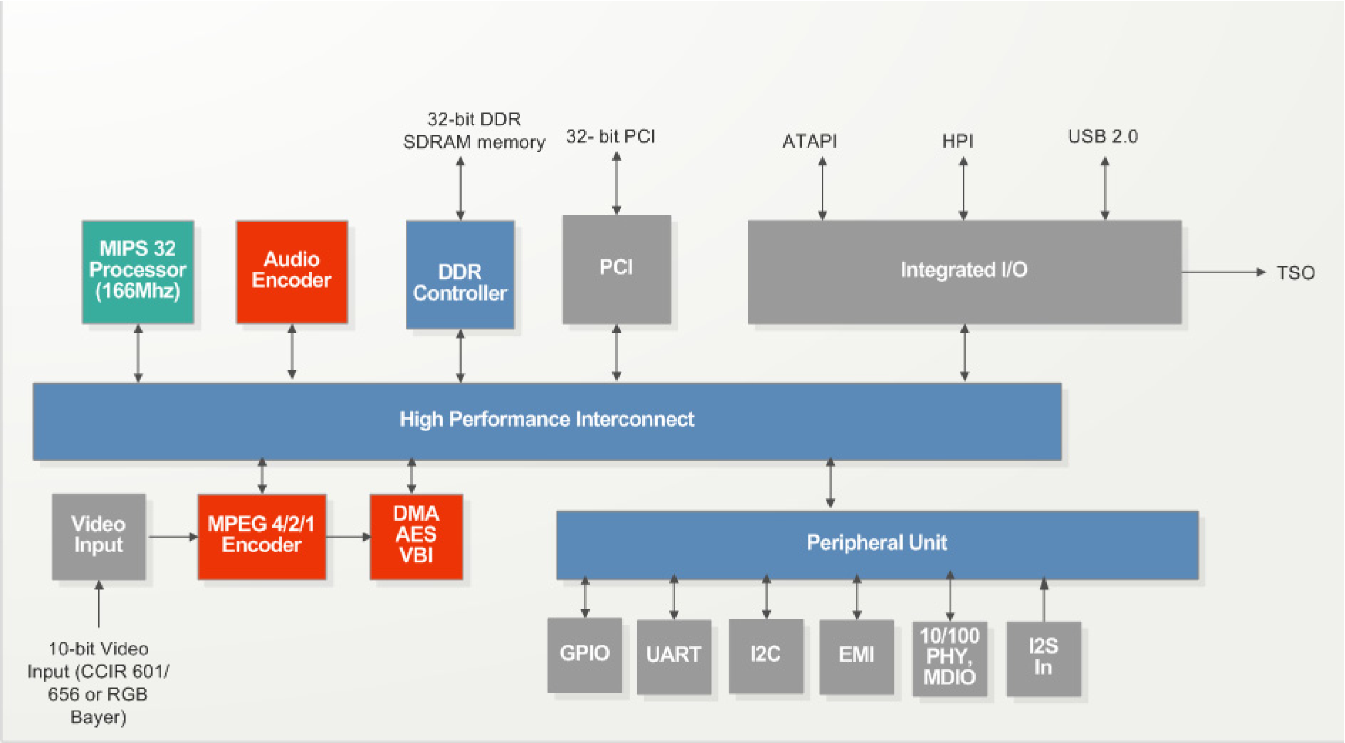


Fig. 1–1: Cypher ESN7108A MPEG Encoder Block Diagram

1.1. Feature

Peripheral Unit

- PCI
- UART
- GPIO
- I^{2}C
- EMI
- Ethernet MAC (EMAC) with MDIO
- Audio Input Interface
- HPI
- ATAPI-5 controller
- Transport Stream Output (TSO)

Video Encoder

- Supports industry-standard MPEG encoding
  - MPEG-4 ISO/IEC 14496-2 Advanced Simple Profile [1]
  - MPEG-2 ISO/IEC 13818-2 MP & ML
  - MPEG-1 ISO/IEC 11172-2
- H.263/ H.263+
- MJPEG
- Raw Capture (RGB565, RGB16, RGB24, RGB32)
- Progressive and Interlace coding

Audio Encoder

- µ-law/A-law (G.711 Voice Codec)
- Dolby Digital (AC-3) (2-channels)
- MPEG-1, and -2, Layers I and II

Video Input

- CCIR-601/656 YUV 4:2:2 input in both progressive and interlaced formats
- Input resolution from 64 x 64 to 720 x 576 in 16-pixel increments. Frame rates up to 30 fps (full D1)
- Supports NTSC and PAL video formats
  - NTSC: 720 x 480 @ 30 frames per second (FPS)
  - NTSC: 720 x 240 @ 60 frames per second (FPS)
  - PAL: 720 x 576 @ 25 frames per second (FPS)
  - PAL: 720 x 288 @ 50 frames per second (FPS)

Audio Interfaces

- Supports industry standard I^{2}S audio input format in both slave and master mode and contains a built-in clock divider
- A/V sync function allows for accurate synchronization of video and audio streams

On Screen Display

- Allows the user to embed any combination of letters, numbers, or graphics onto the image prior to encoding
- Implemented on a YCbCr 4:2:0 video source prior to encoding
- Eight-level alpha blending
- Foreground color defined in YCbCr space with 8-bits per pixel on each channel

Security

- AES 128-, 192-, or 256-bit encryption and decryption in ECB, CBC, CTR, CFB and OFB modes

Flexible Downscaling

- Flexible downscaling mechanism allows for the scaling of an image at any horizontal or vertical ratio between 1 and 63

Test/Debug

- JTAG/MIPS EJTAG

[1] No AC Prediction, GMC, RVLC, Data Partition, Direct mode, 4 motion vectors, Quarter Pixel Interpolation

Image Processing

– Programmable median, low-pass, and edge enhancement filtering

– Cropping function to any number of horizontal and/ or vertical lines

– CCD/CMOS sensor support with noise filtering and image enhancement

– Adaptive 10-bit RGB Bayer reconstruction

– Gamma correction

– Motion adaptive deinterlacing functions

– Horizontal 5-tap filtering in 10-bit RGB Bayer input mode

– 3-tap horizontal and vertical filtering in all input modes

– Hardware statistics support for auto exposure (AE) and auto white balance (AWB)

Mechanical Specifications

– 484 ball grid array package (23 mm x 23 mm)

Product Offerings

Table 1–1: Product Offerings

Product	AES Encryption
7108	Yes
7118	No

1.2. Applications

The rich feature set of the Cypher ESN7108A makes it an ideal choice for several applications in PC, home entertainment, IP cameras, wireless TV, and video surveillance systems.

Figure 1–2 shows the Cypher ESN7108A implemented in a half-PVR application. In this application the video decoder interfaces to the ESN7108A via a CCIR-656/ 601 compliant interface. The encoded video stream can be output over USB 2.0, 10/100 Ethernet, or the PCI bus interfaced to a wireless adapter.

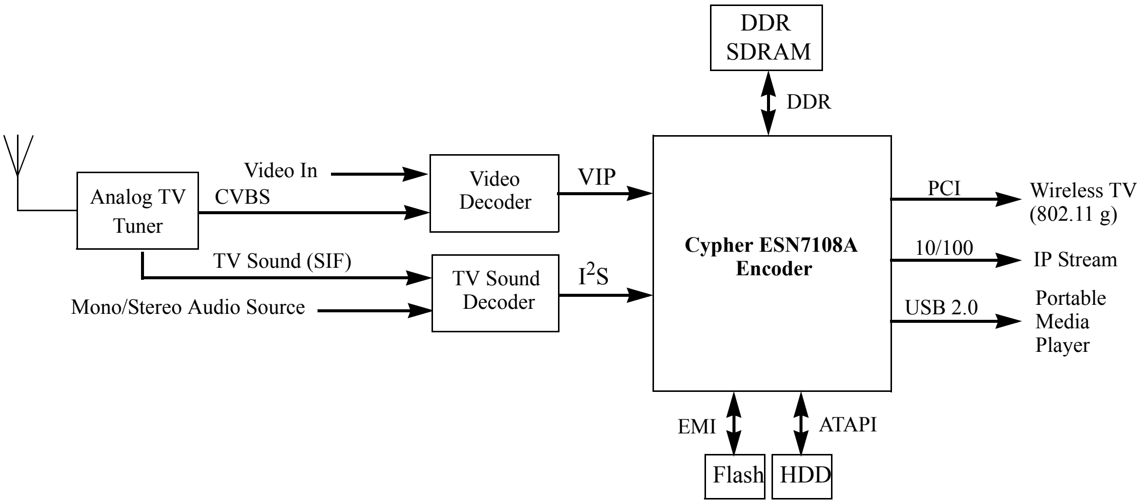


Fig. 1–2: Network Half-PVR Application

Figure 1–3 shows the Cypher ESN7108A implemented in IPCAM application. In this application CCD input data is transferred to the ESN7108A via a CCIR-656/ 601 compliant interface. Encoded data is output through PCI based Wireless LAN adapter, or the on-chip Ethernet MAC interface, which connects to an external 10/100 PHY device.

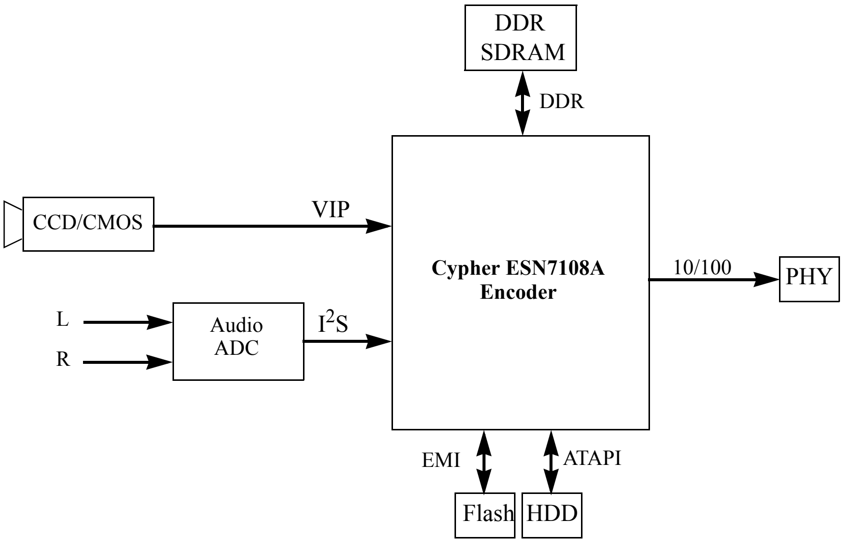


Fig. 1–3: IPCAM Application

In the PCI application the video encoder interfaces to the ESN7108A via a CCIR-656/601 compliant interface. Audio signals are transferred via the I^{2}S interface. Encoded data is transferred via the PCI bus. Note that the PCI bus on the ESN7108A is 3.3V compliant. Legacy systems with a 5V PCI bus will require an external 5V tolerant buffer.

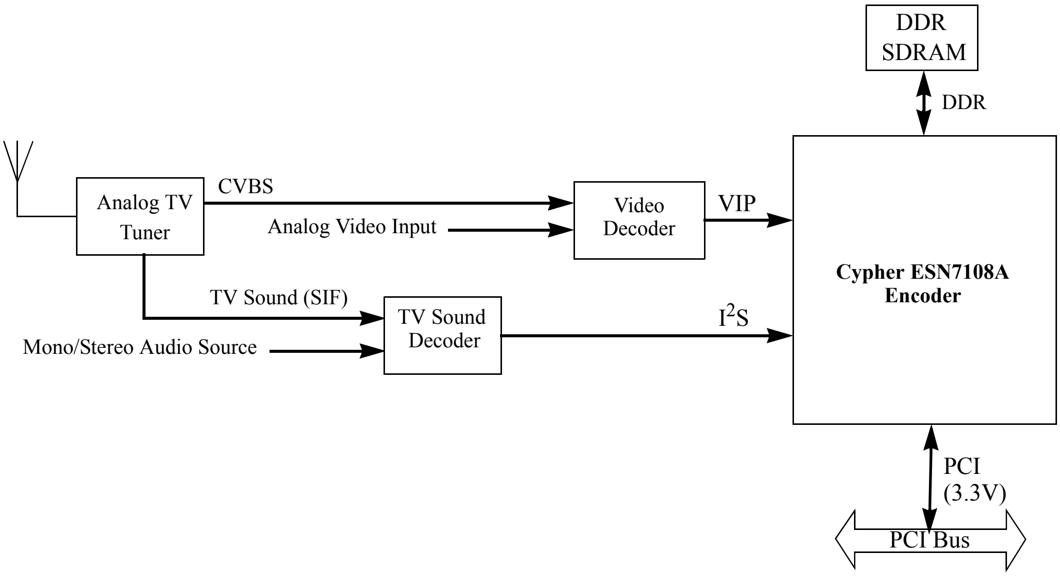


Fig. 1–4: PCI Card Application

2. System Description

2.1. MPEG-4/2/1 Encoder

The MPEG-4/2/1 encoder includes the following logic blocks:

- Section 2.1.1., "Video Input Interface"
- Section 2.1.2., "Video Input Processor (VIP)"
- Section 2.1.4., "Image Input Processor (IIP)"
- Section 2.1.5., "Vertical Blanking Interval (VBI)"
- Section 2.1.6., "Motion Estimation and Compensation (MEC)"
- Section 2.1.7., "Discrete Cosine Transfer/Inverse Discrete Cosine Transfer (DCT/IDCT)"
- Section 2.1.8., "Quantizer/Dequantizer (Q/DQ)"
- Section 2.1.9., "VLC"

Each of these blocks is described in the following subsections:

2.1.1. Video Input Interface

Raw video data is input through a 10-bit parallel interface that is clocked by VID_PCLK (video in pixel clock). The maximum VID_PCLK rate is 27 MHz. The HREF (Horizontal Reference) and VREF (Vertical Reference) signals are used to indicate the beginning of valid data in a line or field. The VID_VALID signal can be used to indicate the valid data during the VREF and HREF active time. In interlace input mode, the VID_FID signal is used to indicate the field sequence.

The ESN7108A supports the following video formats:

- 8/10 bit CCIR-601 compatible, YUV 4:2:2 interlaced or progressive input.
- 8/10 bit CCIR-656 compatible, YUV 4:2:2 interlaced or progressive input.
- 8/10-bit RGB Bayer with pattern sensor formats.

User-configurable filters can be applied on input video data before compression. These filters include subsample, sub-window, downscaling, and median/low pass/edge enhancement. The internal registers of the VIP module contain the filters' configuration bits. Most CMOS and CCD Sensors or any NTSC/PAL video decoder can be connected to the video input interface without any glue logic.

Comprehensive image pre-processing functions are provided in RGB Bayer input mode, including:

- Adaptive Demosaic
- Color correction
- RGB gain and offset adjustment
- Separated RGB gamma correction curve function
- Contrast and Brightness control
- Hue and Saturation adjustment
- 5 tap horizontal de-noise filter
- AE (Automatic Exposure) and AWB (Automatic White Balance) statistics

The ESN7108A supports flexible video data input formats, including progressive/interlaced YUV 4:2:2 and four types of RGB Bayer as shown in the figures below. Optional HREF (Horizontal Reference) and VREF (Vertical Reference) signals can be used to indicate the beginning of valid data in a line or field. In interlaced input mode, the VID_FID signal is used to indicate the field sequence. These reference/identifying signals can be omitted by using ITU656 mode, in which case the timing reference is embedded into the video stream. The VID_VALID signal can be used to indicate the valid data in case that the timing reference signals are inadequate to indicate the validation of video data. The polarity of VID_HREF, VID_VREF, VID_FID and VID_VALID is programmable.

The ESN7108A can accept up to 10-bit inputs in RGB Bayer mode. When the video source is 8-bit, they should be connected to the 8 MSBs of the 10-bit VID_PDATA bus. In this case, the 2 LSBs can be connected to either high or low level.

The ESN7108A can also work in master mode to drive the VID_VREF and VID_HREF signals, in conjunction with a CCD timing generator (which is working in slave mode).

2.1.1.1. Horizontal Timing

8-bit YUV 4:2:2 Mode

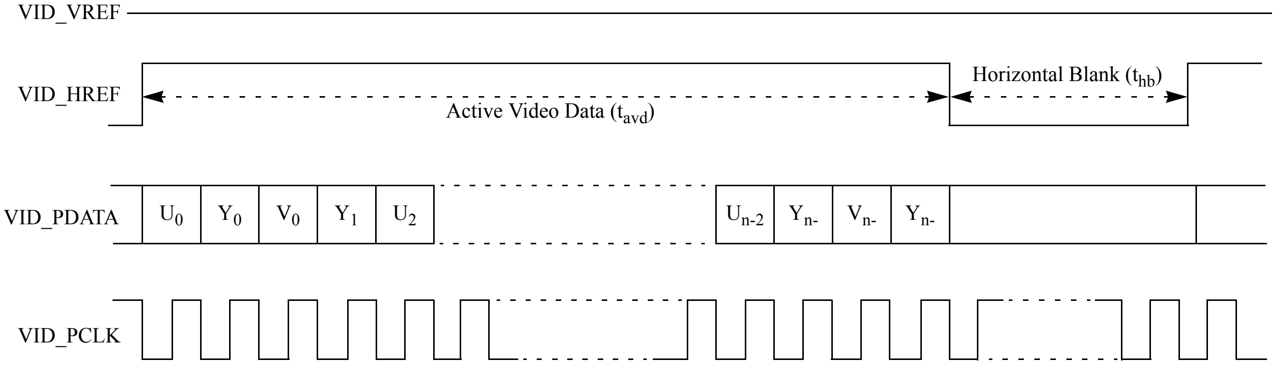

VID_VREF
VID_HREF
Horizontal Blank (t_{hb})
Active Video Data (t_{avd})
VID_PDATA
U_0 Y_0 V_0 Y_1 U_2
U_{n-2} Y_{n-} V_{n-} Y_{n-}
VID_PCLK

Fig. 2–1: 8-bit YUV 4:2:2 Mode

Note: To insure proper operation of the Cypher ESN7108A, the sensor must be configured as follows: $t_{hb} \geq 0.1 t_{avd}$.

RGB Bayer Mode 1

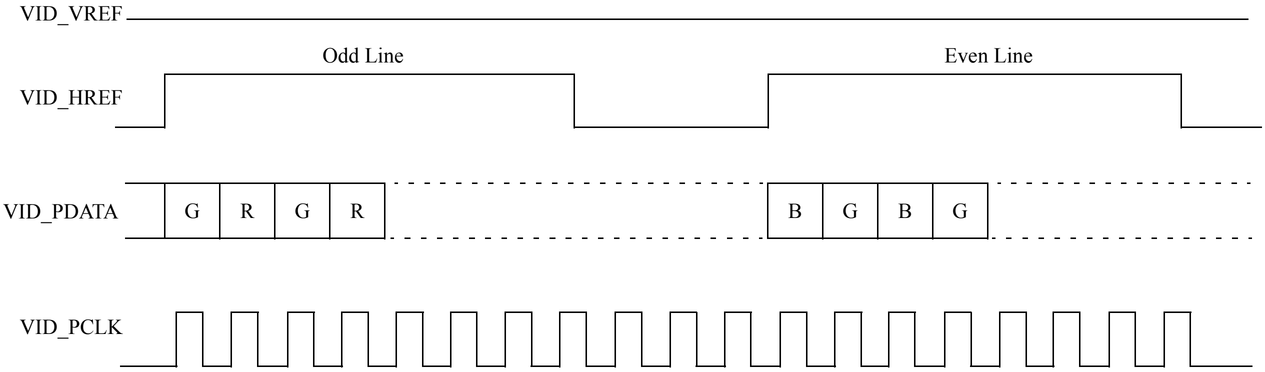


Fig. 2–2: RGB Bayer Mode 1

RGB Bayer Mode 2

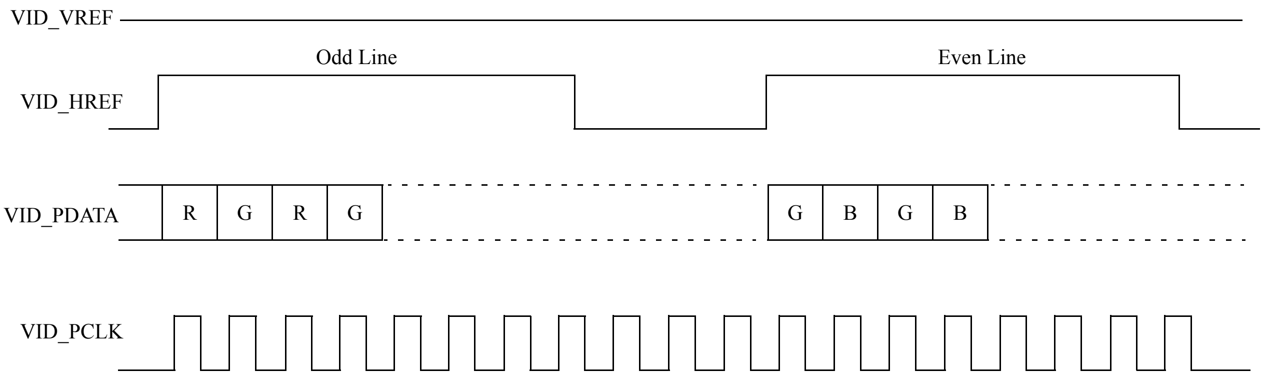


Fig. 2–3: RGB Bayer Mode 2

RGB Bayer Mode 3

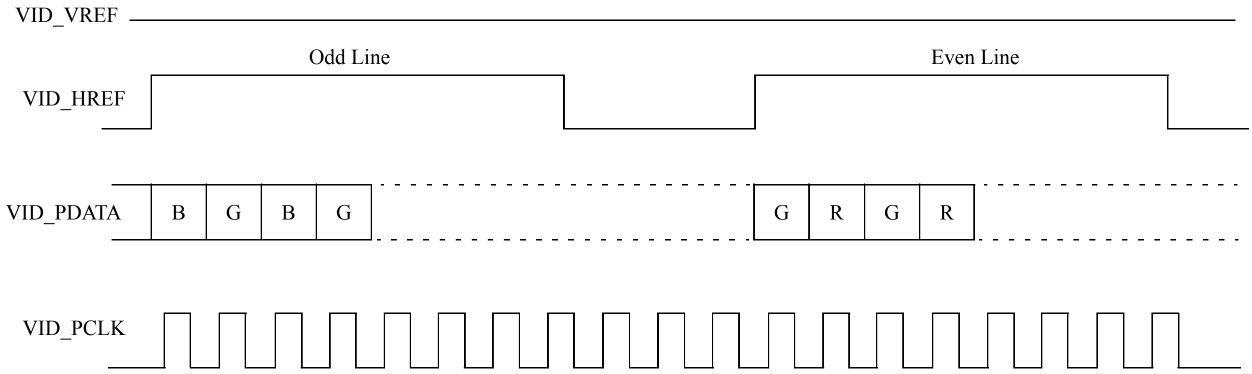


Fig. 2–4: RGB Bayer Mode 3

RGB Bayer Mode 4

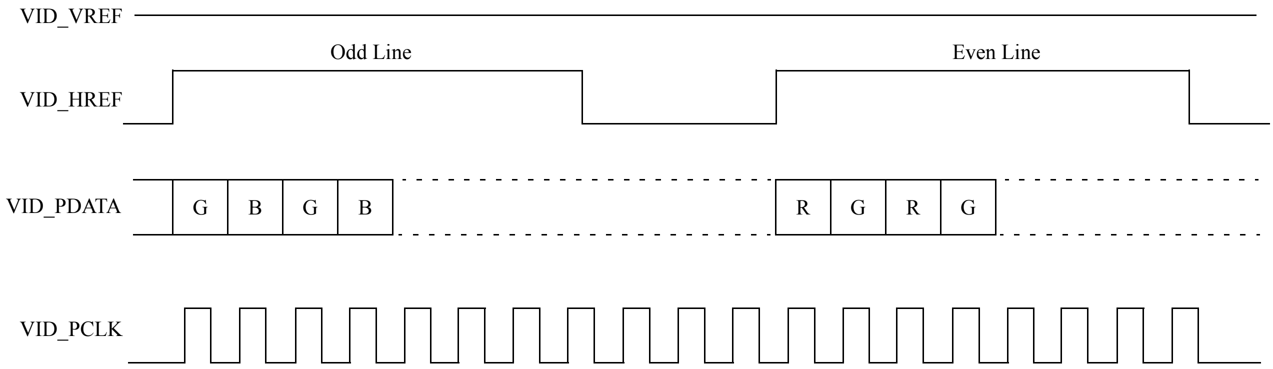


Fig. 2–5: RGB Bayer Mode 4

2.1.1.2. Vertical Timing

Progressive Mode

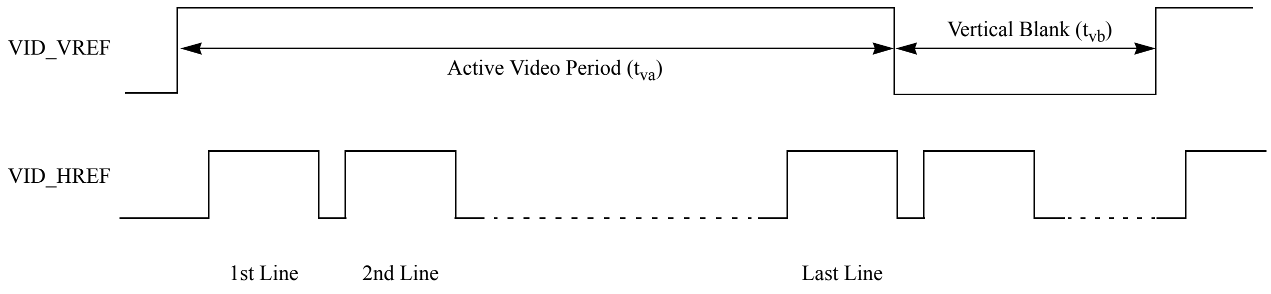


Fig. 2–6: Progressive Mode

Note: To insure proper operation of the Cypher ESN7108A, the sensor must be configured as follows: $t_{vb} \geq 0.05\ t_{va}$.

Interlace Mode (8 Bit YUV 4:2:2 Only)

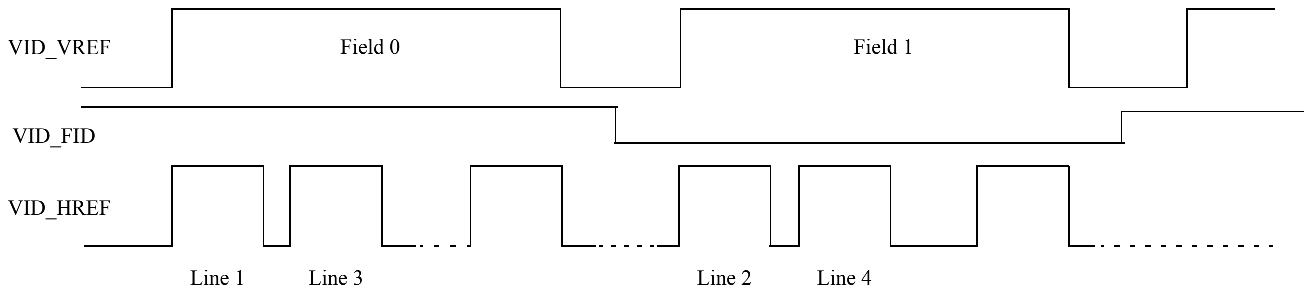


Fig. 2–7: Interlace Mode

For proper operation, the FID should only change at the inactive period of VREF.

ITU656 Input Mode

ITU656 input mode is recommended for applications that have difficulty generating timing reference signals. Since the timing reference code is inserted into the video stream (as shown in following diagram), the external VID_VREF and VID_HREF pins can be left unused. After detecting the timing reference code, the ESN7108A generates the VID_VREF and VID_HREF signals for internal use.

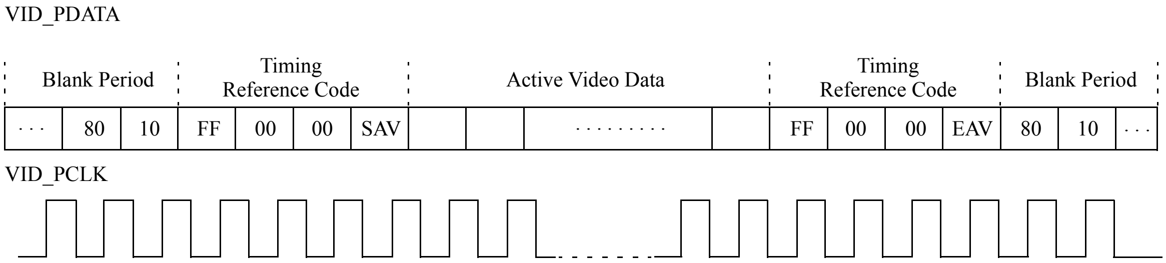


Fig. 2–8: ITU656 Input Mode

The SAV/EAV timing reference codes define the start and end of a valid video data region.

SAV: Start of Active Video
EAV: End of Active Video

Table 2–1: SAV/EAV Format

Bit7	**Bit6** **(F)**	**Bit5** **(V)**	**Bit4** **(H)**	**Bit3** **(P3)**	**Bit2** **(P2)**	**Bit1** **(P1)**	**Bit0** **(P0)**
1	1st field: F=0 2nd field: F=1	VBI: V=1 Active Video: V=0	H=0: in SAV H=1: in EAV	Protection bits See <CrossRef>Table 2–2 below for encoding.			

Table 2–2: Protection Bits in the Timing Reference Signal

T	**F**	**V**	**H**	**P3**	**P2**	**P1**	**P0**	**Hex Value**
1	0	0	0	0	0	0	0	80
1	0	0	1	1	1	0	1	9D
1	0	1	0	1	0	1	1	AB
1	0	1	1	0	1	1	0	B6
1	1	0	0	0	1	1	1	C7
1	1	0	1	1	0	1	0	DA
1	1	1	0	1	1	0	0	EC
1	1	1	1	0	0	0	1	F1

In some cases, the video source may not provide the correct protection bits listed in the above table as P3 to P0. The ESN7108A can be configured to omit those protection bits but and only process the F, V and H bits in the timing reference code.

The ESN7108A can also accept 10-bit ITU656 input, where the timing reference code is in the order of 0x3FF, 0x000, 0x000 and then (SAV or EAV). The 8 MSBs of SAV/EAV are the same as in 8-bit mode, while the 2 LSBs will be zeros.

Video Input with VID_VALID Signal

An optional VID_VALID signal, combined with timing reference signals, can be used to indicate the validation of input video data as shown in figure below.

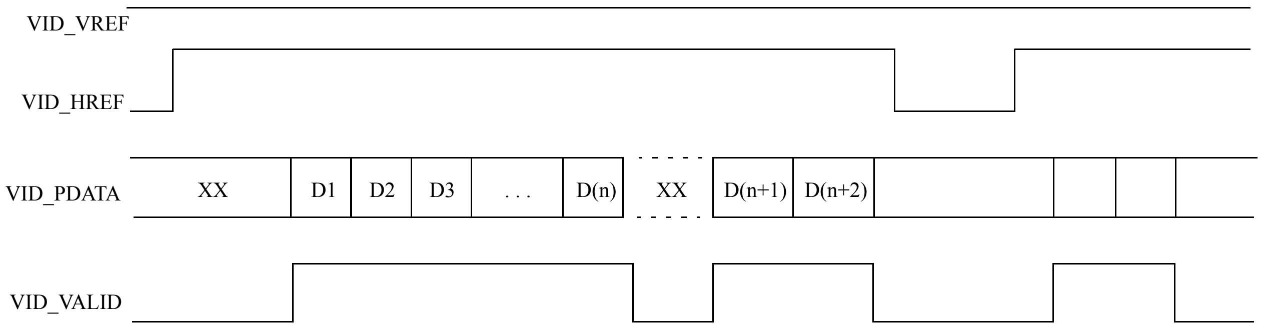


Fig. 2–9: Video Input with VID_VALID Signal

Data is valid only when VREF, HREF and VALID are all active.

Programmable Vertical and Horizontal Timing Generator

The ESN7108A can work in master mode. In this mode, the internal programmable timing generator drives the VID_VREF and VID_HREF pins to synchronize the outside video source with the internal vertical and horizontal timing. A typical application for this mode is to drive an external CCD timing generator.

Video Input Synchronizer

Under certain conditions, the input video stream may become unstable. For instance, when capturing video from a VCR, the output of the TV decoder swings with the playback speed of the VCR. This problem could be serious when going in the forward or reverse directions. The ESN7108A contains an optional video input synchronizer to track the video input. Upon losing synchronization, it shuts off the video input until it can be resynchronized to avoid internal errors of the encoder.

2.1.2. Video Input Processor (VIP)

The Video Input Processor (VIP) is responsible for interfacing to the external digital video source. It also enables video format conversion, RGB Bayer to RGB conversion, color space conversion, de-interlacing, frame sub-sampling, special sub-sampling, and video frame buffering.

The VIP block outputs a YUV 4:2:0 video stream, which is buffered and written into external DRAM by a DMA controller. The VIP also outputs a Y-only 16-to-1 sub-sampled frame for motion estimation and compensation. This frame is also written to the DRAM buffer. According to the GOP structure algorithm, the DRAM can buffer up to 5 input pictures.

2.1.2.1. Horizontal/Vertical (HV) Sync Generator

In most circumstances the ESN7108A acts as a slave device. However, the ESN7108A can also operate in master mode. This typically occurs when the ESN7108A is connected to a sensor or other device that requires it to be a master. In master mode, the internal programmable timing generator drives the horizontal sync signal (HD) and vertical sync signal (VD) onto the VREF and HREF pins respectively to synchronize the outside video source with the internal vertical and horizontal timing. The typical application for this mode is to drive an outside CCD timing generator working in slave mode.

2.1.2.2. 656-Header Decipherer

Refer to Section 2.1.1.2. "Vertical Timing" on page 13.

2.1.2.3. VBI Data Extractor

The ESN7108A extracts Vertical Blanking Internal (VBI) data from the ITU656 video stream. Refer to Section 2.1.5., "Vertical Blanking Interval (VBI)" for more information.

2.1.2.4. Flexible Down Scaling Control and Subsampling

The Cypher ESN7108A contains a flexible downscaling mechanism allows for the scaling of an image at any horizontal or vertical ratio between 1 and 63.

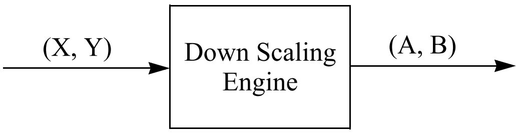


Fig. 2–10: Subsampling Engine Inputs and Outputs

In the Fig. 2–10, the X value above is multiplied by the downscaling ratio to determine the output value A. The Y value above is multiple by the downscaling ratio to determine the output value B. Since this is a downscaling unit, the following limitations exist:

$m_H < n_H$
$m_V < n_V$
n_H, n_V, m_H, and m_V are an integer values less than 64

In addition, the subsampling function works in conjunction with the flexible down scaling unit. If sub-sampling is enabled, the input data is reduced by a factor of 2 prior to entering the downscaling unit.

Therefore,

$A = m_H/n_H \times X$ (subsampling disabled)
$A = m_H/n_H \times X/2$ (subsampling enabled)
$B = m_V/n_V \times Y$ (subsampling disabled)
$B = m_V/n_V \times Y/2$ (subsampling enabled)

For example, assume that X = 720 and Y = 360. The desired output is A = 480 and B = 240. In this case, the horizontal ratio would be calculated as follows:

$A/X = m_H/n_H$
$480/720 = 2/3$
$m_H = 2$
$n_H = 3$

The vertical ratio would be calculated as follows:

$B/Y = m_V/n_V$
$240/360 = 2/3$
$m_V = 2$
$n_V = 3$

2.1.2.5. Asynchronous FIFO

The 8-pixel asynchronous FIFO is used to synchronize the sensor clock domain to the main clock domain of the ESN7108A. The input side of the FIFO is in sensor clock domain, and the output side is synchronized to the main clock domain.

2.1.2.6. Synchronizer

This functional block is used to synchronize the FIFO output to the main clock domain of the ESN7108A.

2.1.2.7. Cropping

Cropping is used when the number of pixels in the image is not a multiple of 16, or if only a portion of the image is required.

The sensor image must be cropped into a size of a multiple of 16 before encoding. Users can get any part of the sensor image with any size (multiple of 16) by programming the offset and size of the cropped image. The cropping parameters are shown in Fig. 2–11.

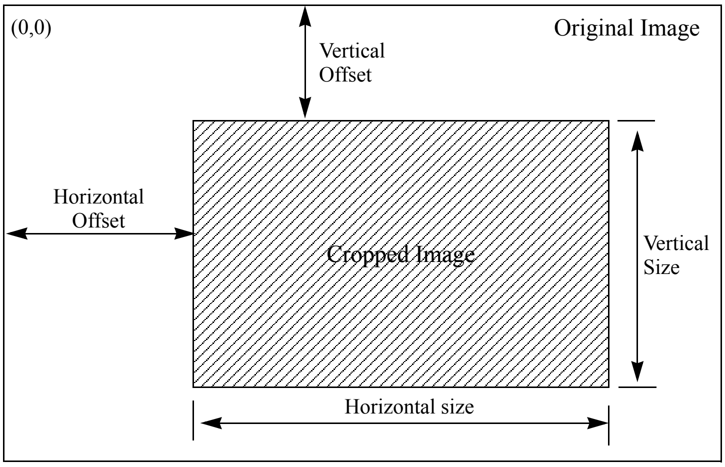


Fig. 2–11: Cropping Function

2.1.2.8. Re-synchronization

This functional block can detect errors in the sync signals from sensor. It can also fix these errors and keep the integrity of the image data.

2.1.3. IIP (Image Input Processor)

The IIP block converts RGB Bayer input into YCbCr 4:2:2 progressive, it also provides some image processing functions. Refer to Section 2.1.4., "Image Input Processor (IIP)" for more information.

The IIP should be bypassed when the sensor input is YCbCr 4:2:2 progressive or YCbCr 4:2:2 interlace.

2.1.3.1. Drop frame

This functional block allows users to lower the frame rate of sensor by dropping some frames in the input data.

2.1.3.2. Input FIFO

The 1024x32b input FIFO can buffer at most 4 rows of RGB Bayer or 2 rows of YCbCr 4:2:2 image data.

2.1.3.3. YCbCr 4:2:2 to 4:2:0

This block processes the YCbCr 4:2:2 sensor input data.

For YCbCr 4:2:2 progressive input, the data will be converted into YCbCr 4:2:0 progressive.

For YCbCr 4:2:2 interlace input, there are two options. One is to keep the input data untouched and store them into DRAM. In this case, the output of VIP is YCbCr 4:2:2, and the data will be processed by the motion-adaptive de-interlace module in the encoding part of the chip. The other option is to use the de-interlace function built in this block. One of the fields, either top field or bottom field will be dropped, and the other field is interpolated to reproduce the whole frame. In this case, the output of VIP is YCbCr 4:2:0 progressive.

When sensor input is RGB Bayer and IIP is bypassed, the input of this module is still RGB Bayer, so this block should be bypassed. If sensor is RGB Bayer and IIP is enabled, then the input of this block is YCbCr 4:2:2 progressive, so it should not be bypassed.

2.1.3.4. RGB to YCbCr converter

This block converts the RGB Bayer input into YCbCr 4:2:0 progressive. It supports 4 types of Bayer pattern: RG, GR. BG and GB.

When sensor input is YCbCr progressive or YCbCr interlace, this block should be bypassed.

When user choose IIP to perform the RGB to YCbCr conversion (recommended), this block should also be bypassed.

Comparing with this converter, IIP provide better quality and more image processing functions. The ESN7108A keeps this block only to be backward compatible with GO7007, which has no IIP.

2.1.3.5. Output FIFO

The 1024x64b output FIFO can buffer at most 4 rows of output image data.

2.1.3.6. Filter

The ESN7108A contains a multi-state tap filtering circuit for input data. For 10-bit RGB Bayer input data, a 5-tap horizontal filter is provided. This is followed by a second stage filter that provides 3-tap horizontal and vertical filtering.

For YUV input data, the 5-tap filter is bypassed and only the 3-tap horizontal/vertical filter is used. This concept is shown in figure below.

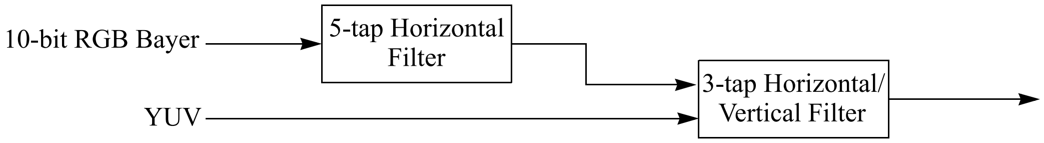


Fig. 2–12: 5-Stage Tap Filtering for 10-bit RGB Bayer Input Data

2.1.4. Image Input Processor (IIP)

Features provided by the Image Input Processor (IIP) allow the Cypher ESN7108A to work with high quality image sensors without requiring an on-chip image processor such as the Hynix HV7131, the Motorola MCM20014, and numerous CCDs, etc. These sensors only provide RGB Bayer data output and have no on-chip image processing such as demosaic, color correction, gamma correction and color conversion (RGB to YUV), etc, to get optimized images.

2.1.4.1. Features

- Adaptive Demosaic for less zigzags and false colors.
- Color correction matrix for more accurate and richer colors.
- RGB gain and offset adjustment (for color balance and black level adjustment).
- Separated RGB gamma correction curve function.
- Contrast and Brightness control.
- Hue and Saturation adjustment.
- 5-tap horizontal de-noise filter.
- AE and AWB statistics.

2.1.4.2. Adaptive Demosaic

The missed color information in the Bayer input are filled by the edge-adaptive demosaic algorithm, providing enhanced sharpness and less false color. The algorithm is based on 5x5 neighbors, as shown in figure below.

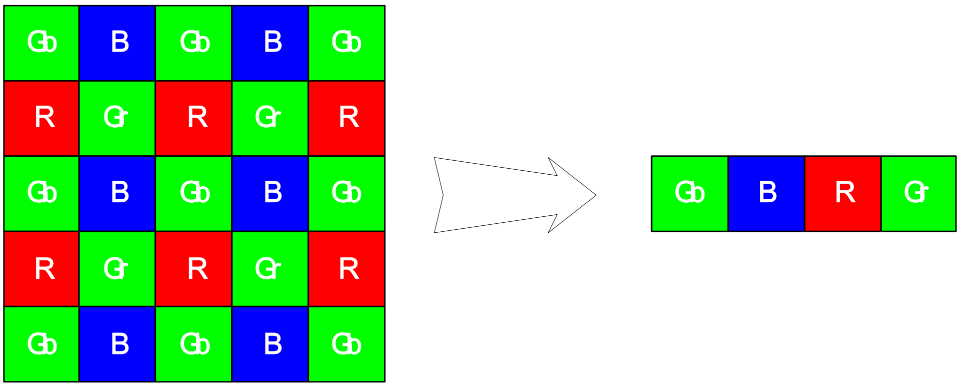


Fig. 2–13: Adaptive Demosaic

The algorithm treats the Bayer pattern as four separate colors so as to support the CMYG image sensor.

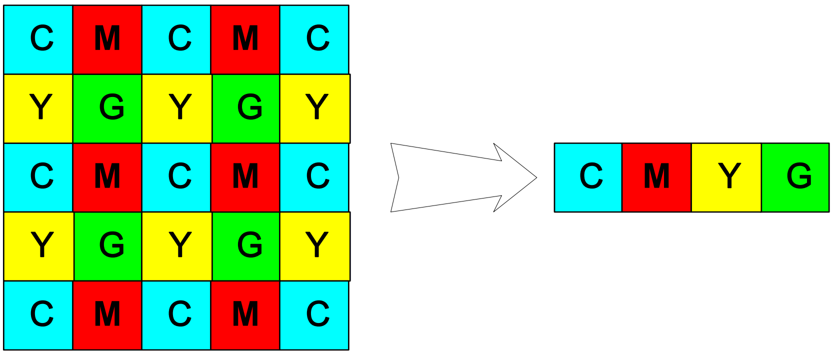


Fig. 2–14: CMYG Bayer Pattern

2.1.4.3. Color Correction

4x3 color correction matrix provides following functions:

- Transferring results from Demosaic module to RGB color space.
- Eliminating cross talking of color filters on sensors.

$$\begin{bmatrix} r & g & b \end{bmatrix} \equiv \begin{bmatrix} c_1 & c_2 & c_3 c_4 \end{bmatrix} x \begin{bmatrix} ccm_{11} & ccm_{12} & ccm_{13} \\ ccm_{21} & ccm_{22} & ccm_{23} \\ ccm_{31} & ccm_{32} & ccm_{33} \\ ccm_{41} & ccm_{42} & ccm_{43} \end{bmatrix}$$

where [c_1, c_2, c_3, c_4] is the result from the Demosaic module. For instance, for the RGB Bayer pattern in the above is [c_1, c_2, c_3, c_4] = [G, B, R, G], then a basic Color Correction Matrix is:

$$MatrixCCM = \begin{bmatrix} 0x0 & 0x80 & 0x0 \\ 0x0 & 0x0 & 0x100 \\ 0x100 & 0x0 & 0x0 \\ 0x0 & 0x80 & 0x0 \end{bmatrix}$$

The matrix coefficients ccm_{ij} are programmable from: -511/256 to 511/256.

2.1.4.4. Gain and Offset Adjustments

This module provides gain and offset adjustments for white balance and black level calibration.

Gains are programmable from 1023/256 to 1/256. Offsets are programmable from -127 to +127.

2.1.4.5. Separate RGB Gamma Correction

There are three separate gamma correction curves for RGB channels. Each of these curves is customer programmable.

2.1.4.6. RGB to YCrCb

Use CCIR601 standard matrix to convert RGB to YCrCb color space.

2.1.4.7. YUV Adjustments

This module provides the following image adjustments:

– Saturation: 127/64 ~ 0.

– Hue: -31 degree ~ +31 degree.

– Brightness: -127 ~ +127.

– Contrast: 127/64 ~ 1/64.

2.1.4.8. De-Noise Filter

The ESN7108A contains a multi-state tap filtering circuit for input data. For 10-bit RGB Bayer input data, a 5-tap horizontal filter is provided. This is followed by a second stage filter that provides 3-tap horizontal and vertical filtering.

For YUV input data, the 5-tap filter is bypassed and only the 3-tap horizontal/vertical filter is used. This concept is shown in Fig. 2–12.

2.1.4.9. YUV 4:4:4 to YUV 4:2:2

Converts YUV 4:4:4 to YUV 4:2:2 format for internal use.

2.1.4.10.AE and AWB Statistics

AE (Automatic Exposure) and AWB (Automatic White Balance) play a very important role in image sensor signal processing. The IIP is able to calculate statistics data for both AE and AWB on a frame basis. The internal processor (MIPS) or external host can then use the data for AE and AWB control. Configuration of the IIP and image sensor are adjusted to provide optimum exposure and white balance. The data flow is shown in Figure 2–15:

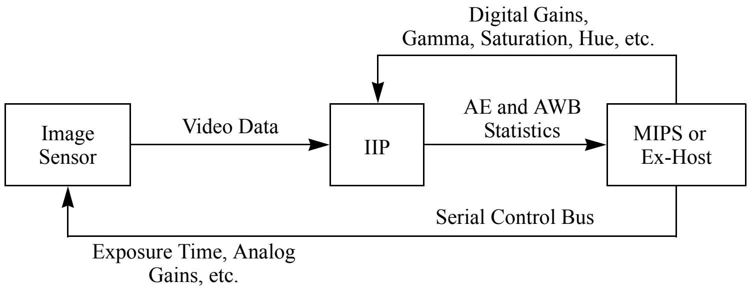


Fig. 2–15: AE and AWB Data Flow

2.1.5. Vertical Blanking Interval (VBI)

The Vertical Blanking Interval (VBI) protocol is used to transfer data during the vertical blanking period. The VBI block supports the following formats.

- ANC
- Line number. In this mode the line number register is first written, then the value of the line number that contains the valid VBI data waits for that particular line number to occur, then transfers the data corresponding to the line numbers indicated by the register value as VBI data.
- T bit. This mode indicates that the module functions in the T bit mode. The header is filtered out and all remaining data is passed on as valid VBI data. In the case where the header comes between the data, the data is again filtered out and not validated as valid VBI data.

2.1.5.1. ANC Mode Line Format

VBI data is transferred in a specific sequence. This sequence is preceded by an ANC header in the vertical blanking period of the input video stream as shown in Figure 2–16.

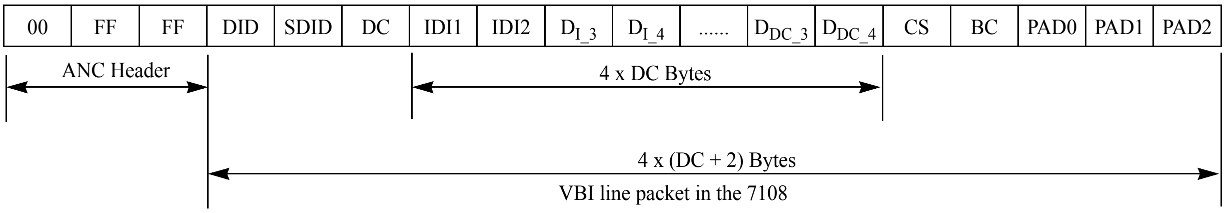


Fig. 2–16: ANC Mode Line Format

The ESN7108A strips off the ANC header information before storing the VBI line packet to the internal VBI FIFO. Three padding bytes are appended to the end of the packet to make the packet doubleword-aligned. In Figure 2–16, the following abbreviations are used:

- DID, SDID — These bits are programmed by the TV decoder to indicate the start of a vertical blanking sequence.
- DC — Doubleword count. This information is inserted by the ESN7108A based on information received from the decoder.
- IDI1, IDI2, D_{I_3}, D_{I_4}, D_{DC_3}, D_{DC_4} — This sequence of bytes indicates the data portion of the VBI. Since the value in DC indicates the number of 32-bit doublewords, the number of bytes transferred in the data field is equivalent to four times the value in DC, as shown in Figure 2–16.
- CS — Checksum
- BC — Byte Count
- PAD0, PAD1, PAD2 — Three zero-padded bytes added to the end of the VBI line to make the packet doubleword-aligned.

2.1.5.2. Line Number Mode

This mode specifies the VBI line number enable bits during the vertical blanking period. Data is captured from the corresponding VBI-enabled line numbers.

At the trailing edge of the vertical blanking signal, an internal line count register is loaded with the contents of either the field 1 register or the field 2 register. The counter is decremented with each horizontal blank. When the counter reaches zero, VBI data corresponding to the VBI line number bit map register is captured and sent to the Color Conversion Unit (CCU).

.

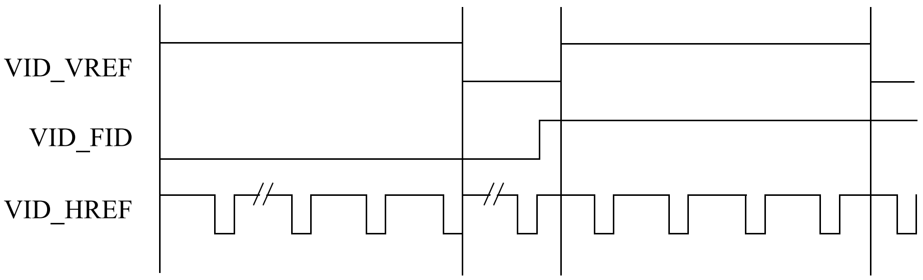


Fig. 2–17: VBI Line Number Mode Data Capture

The Line Number mode is enabled by setting the following registers.

– Field 1 VBI Line Number Bit Map register

– Field 2 VBI Line Number Bit Map register

– Field 1 Line Count register

– Field 2 Line Count register

– VBI Operating Mode register

2.1.5.3. T-bit Mode

This mode specifies capturing VBI data between headers. The data between SAV (Start of Active Video) and EAV (End of Active Video) during the horizontal and vertical blanking periods is captured and passed on to the CCU. Since the SAV and EAV bytes can contain different values, the VBI_EAV_SAV_BYTES register (0x417B) is used to program the EAV and SAV fields. Four combinations of SAV and EAV values can be used for VBI capture as shown in Table 2–3.

Table 2–3: EAV and SAV Field Combinations

Combination	Header Preceding SAV Field	SAV Field	Header Preceding EAV Field	EAV Field
1	0xFF0000	1 a0 a1 0 x x x x	0xFF0000	1 a2 a3 1 x x x x
2	0xFF0000	1 b0 b1 0 x x x x	0xFF0000	1 b2 b3 1 x x x x
3	0xFF0000	1 c0 c1 0 x x x x	0xFF0000	1 c2 c3 1 x x x x
4	0xFF0000	1 d0 d1 0 x x x x	0xFF0000	1 d2 d3 1 x x x x

As shown in Table 2–3, a value of 0xFF0000 is followed by the SAV byte. At the end of the VBI data, another value of 0xFF0000 is detected, followed by the EAV byte. Only the data between the headers is sent to the CCU. Note that if the SKIP_656 protect bit in the VBI Enable register (0x4031) is cleared, the protection bits are checked for correctness of the header and the header is ignored if an error is detected.

The following diagram shows the data format in T-bit mode.

FF	00	00	TS	D0	------------------------------------	D1439	FF	00	00	TE

VBI data is T-bit mode is marked by leading and trailing headers. The data between the headers is captured as VBI data and can have a maximum length of 1440 bytes per line in CCIR 601 as shown above. In the above figure, the first three bytes, FF - 00 - 00 - TS, comprise the leading header, indicating that valid VBI data will be on the following byte. Up to 1440 bytes of VBI data are supported, but can be any value in between. Data capture continues until the trailing header bytes, FF - 00 - 00 - TE, are detected, indicating the end of the VBI transfer.

The TS and TE bytes contain information about the associated VBI data and are defined as follows:

0	F	V	H	P0	P1	P2	P3

Where:

F = field bit
V = vertical bit
H = horizontal bit (0 for TS, 1 for TE)
P0 - P4 = protection bits

The F and V bits for the TS and TE bytes can be programmable to reflect different combinations.

This mode is set by programming the following registers.

- VBI Operating Mode register
- EAV_SAV register

VBI Enable register at address 0x4031 (skip_656_prtct bit).

2.1.6. Motion Estimation and Compensation (MEC)

The Motion Estimation and Compensation (MEC) block comprises the largest portion of the Encoder block. An X-instruction controller and a Y-instruction controller issue the instructions. This specialized architecture provides maximum performance and flexibility.

The streaming buffer provides high-speed video data transfer among the MEC array, DRAM, and DCT/IDCT modules under the coordination of the Scheduler.

A column processor is used to process the estimation results. The column processor seeks the best motion vectors among the Mean Absolute Difference (MAD) and handles out-of-boundary vectors.

The Cypher ESN7108A includes an on-chip cost table to enhance the effectiveness of its motion search engine. Because motion vectors need to be coded in final bit stream, they all bear a “cost” depending on the required bit length to code them. For example, larger motion vector usually entails longer VLC code and therefore higher cost. Cypher ESN7108A incorporates each motion vector's cost before it selects the best match for reference macro-block, leading to better motion search results.

As a significant improvement from Micronas 7008, the ESN7108A chip supports the most sophisticated MPEG2 VLC coding for skipped macro-blocks. This enables very efficient VLC coding for continuously skipped macro-blocks within each frame.

2.1.7. Discrete Cosine Transfer/Inverse Discrete Cosine Transfer (DCT/IDCT)

The DCT/IDCT block performs both the Discrete and Inverse Discrete Cosine Transformation of macro-block based video data. The internal precision of DCT and IDCT is 17 bits, and DCT output is saturated to 8 or 12 bits. Thus, the DCT/IDCT generates little or no mismatches over the ideal floating-point model, and minimizes the DCT mismatch. The DCT algorithm in the ESN7108A is IEEE compliant and DIVX compatible.

Now field DCT coding is supported, which was not available in the previous generation chip.

2.1.8. Quantizer/Dequantizer (Q/DQ)

The QdQ block performs quantization and dequantization based on a zigzag scan over a macro-block. The quantization method can be chosen for a particular compression mode and quantization tables can be programmed to accommodate different compression standards.

2.1.8.1. Quantization Algorithms

The ESN7108A includes individual tables for quantizing the following formats:

- H.263
- MPEG2
- MPEG4
- Non-linear

2.1.8.2. Mismatch Control

The DQ block also contains mismatch control logic. This logic is required by the IEEE to meet MPEG 1 and 2 standards. This block controls the mismatch of data between the encoder and the decoder.

The dequantized value for all the pixels is summed up. When the 64th pixel is reached, hardware checks the resultant sum value. If the sum is odd then do nothing. If the sum is even then make the resultant dequantized value as odd, meaning exclusive of the dequantized value of the 64th pixel with '1'.

2.1.9. VLC

The Variable Length Code encoder generates a 32-bit wide output that is buffered for each macro-block. Programming the VLC internal registers enables VLC block functionality to accommodate a variety of compression standards.

The ESN7108A also includes two new indices have been added to the code book, one for ESC1, and one for ESC2 modes. Having these indices guarantees that the code book will always be able to perform the lookup within 4 clock cycles. If neither of the lookups are valid, the mode is ESC3. Regardless of the complexity of the data stream, the ESN7108A guarantees that the VLC can finish 100% of the time.

The prediction module has been changed to support macro block level quantization or bit rate control within a frame. The scale is 1 to 31 with 1 being the highest quality (largest bit stream and highest bit rate) and 31 being the lowest quality (smallest bit stream and lowest bit rate).

2.2. Peripheral Unit

The Peripheral Unit contains the following interfaces, each of which is described in the following subsections.

- Section 2.2.1., "DMA Controllers"
- Section 2.2.2., "Semaphore Block"
- Section 2.2.3., "PCI Interface"
- Section 2.2.4., "UART Interface"
- Section 2.2.5., "GPIO Interface"
- Section 2.2.6., "I^{2}C Interface"
- Section 2.2.7., "EMI Interface"
- Section 2.2.8., "Ethernet MAC (EMAC) Interface"
- Section 2.2.9., "Audio Input Interface"
- Section 2.2.10., "I^{2}S Audio Input"
- Section 2.2.11., "HPI Interface"
- Section 2.2.12., "ATAPI Interface"

2.2.1. DMA Controllers

The ESN7108A contains two dedicated DMA controllers for the PCI and Ethernet interface, and one peripheral DMA controller (PeriDMA) shared by all other peripherals. Note that the Ethernet port (EMAC) interfaces to both the PeriDMA controller and its own dedicated controller. This is shown by the overlap of DMA controllers to the EMAC interface.

2.2.1.1. Peripheral DMA Controller (PeriDMA)

The PeriDMA controller is shared by the Audio, Transport Stream Output (TSO), and EMAC interfaces and is used to move date to and from memory and the ESN7108A.

PeriDMA Features

The features of the PeriDMA controller are as follows:

- Supports up to 32 DMA channels. Each channel is assigned to a device for Tx or Rx.
- Arbitration: each channel have programmable priority.

- Programmable DMA transfer MTU size.
- Origination/destination address, buffer size, addressing mode, interrupt enable, priority, and other DMA behaviors can be controlled on descriptor basis.
- PeriDMA loads descriptors from a link list structure in the system memory, the pointer to the link list head must be programmed in to periDMA's CSR by firmware.
- The last served descriptor is saved in the CSR SRAM for each channel.
- The PeriDMA controller generates and interrupt at the end-of-block or end-of-link chain, which can be controlled by the descriptor element and the interrupt Mask register.
- Automatic flow control: If enabled, the periDMA polls on the readiness of a link list element. Once the link list is appended a new element (which must be done through change the last element's NEXTPOINTER from NULL to the address of the new available element), the periDMA automatically links to the new element.
- Status update to DDR: After a descriptor is served, a bit is set in the descriptor back in the DDR. If the channel is terminated by any reason, the buffer size of the descriptor is updated.
- Optimized use of system bandwidth: All MTU transfers are internally buffered, no wait state exists in Tx/Rx MTU transfers.
- Global Flush: The processor can write the global Flush register to flush all the channel buffers
- Each channel can support the flush-and-drain function
  - For TX channels, a drain signal will be generated by the DMA engine when it hit the stop bit in descriptor.
  - For Rx Channels, the device can issue a drain or flush signal to ask the DMA engine to either drain or flush the RX channels. DMA engine will return a drain or flush acknowledge.
- Interrupt: DMA engine will generate an interrupt on following events:
  - Stop bit in the descriptor
  - Device issued a flush request
  - Bus error / time out happened
  - The interrupt can be masked of by the processor dynamically.
- Debug support: When a Bus error detected, the DMA engine can enter to debug state, the bus error reason and the state of the DMA engine will be stored. The DMA engine can be resumed by firmware.

2.2.2. Semaphore Block

The Semaphore Array Block (SAB) provides a Multi Processor Communication and Resource Sharing mechanism that guarantees transfers are atomic in nature and provide for mutual exclusion between different processors.

The various functions of SAB are listed below.

- Updates semaphore based on a “Read Set Write Clear” methodology.
- Total of 16 semaphores are supported.
- A semaphore can be interpreted as any resource negotiated by 2 or more processors trying to acquire ownership of that resource.

2.2.3. PCI Interface

The PCI interface in the ESN7108A enables the writing of encoded frames into DDR memory, reading decoded frames from DDR memory, initializing all registers at power-up, and acts as a PCI host for configuring other PCI devices.

The ESN7108A provides:

- 32-bit, 33 MHz PCI operation
- Maximum MTU (Maximum Transfer Unit) of 256 bytes
- Multiple MTU selection for PCI bandwidth sharing.
- One frame transfer per DMA programming
- There are total of 10 DMA channels. 4 receive (from system memory to DDR) channels and 6 transmit (from DDR to system memory) channels.
- PCI Host mode. In this mode, the ESN7108A provides interface logic that allows the on-chip MIPS processor to configure all PCI Devices and PCI Bridge and set up PCI DMA to execute data transfer between DDR memory and external system memory.
- Client mode. In this mode, the ESN7108A permits an external host processor to access its internal register space.
- Bidirectional DMA transfers between DDR memory and external system memory.
- Status Logging, Notify Write, and Interrupt generation mechanism for DMA status tracking.

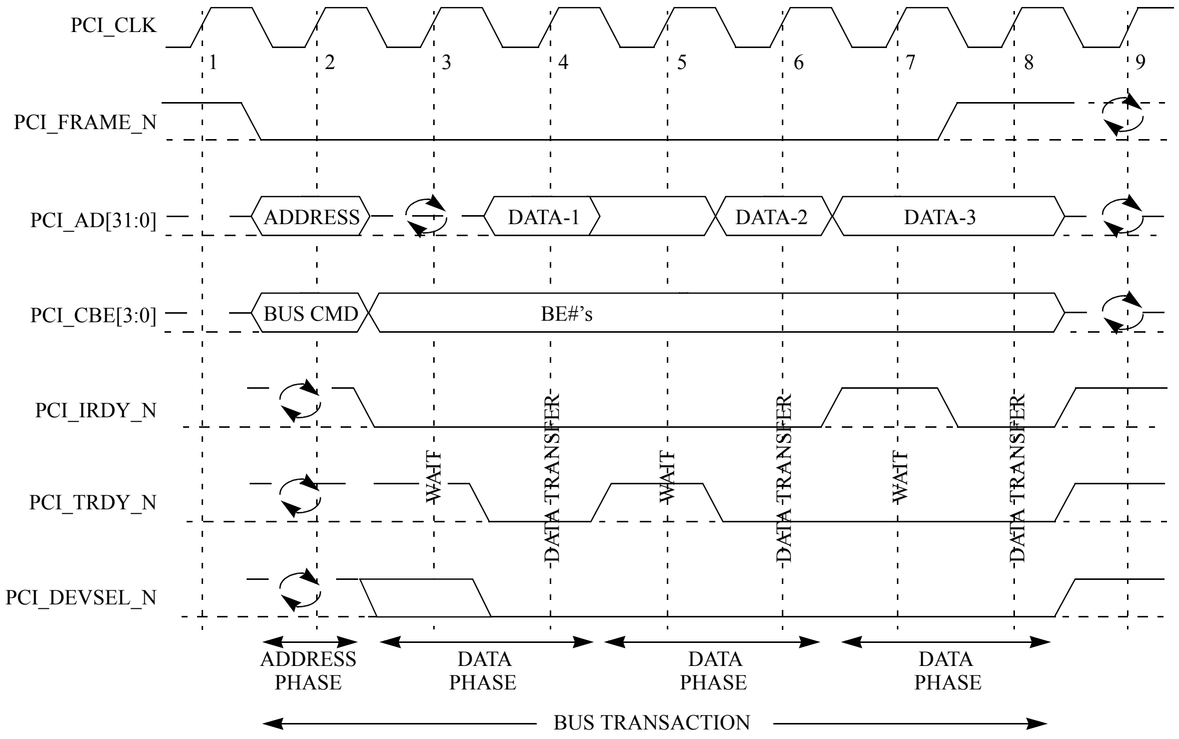


Fig. 2–18: Basic Read Operation

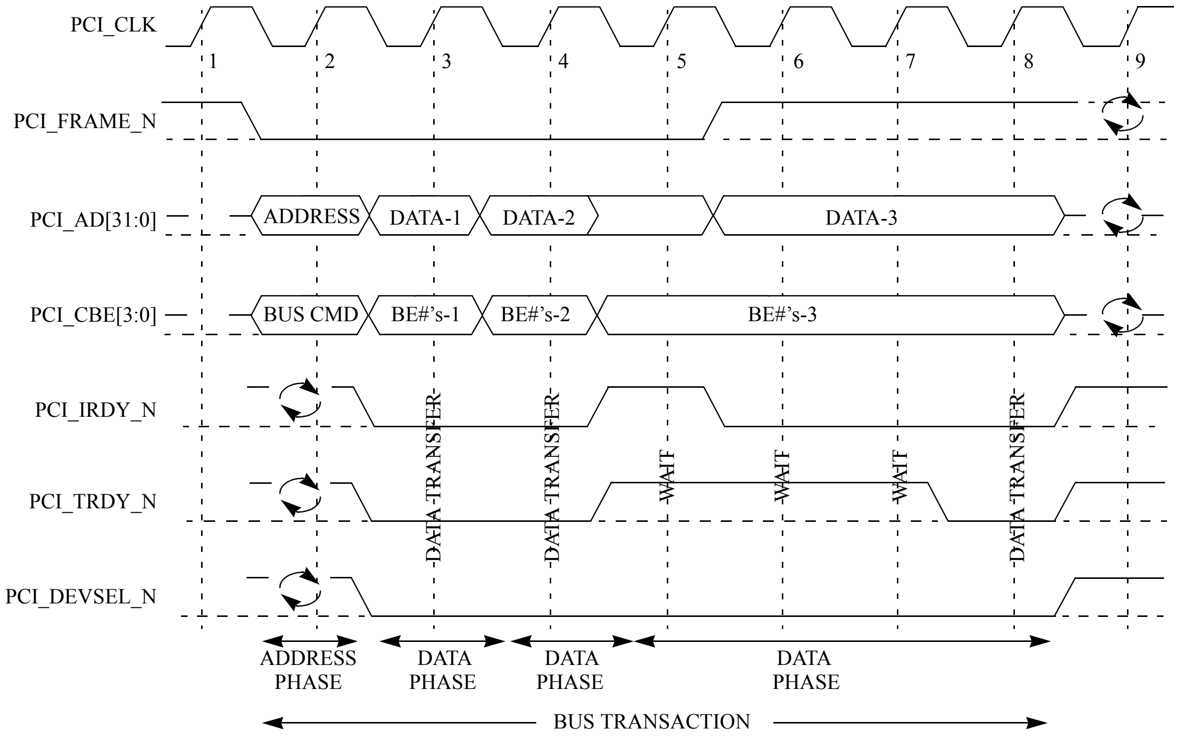


Fig. 2–19: Basic Write Operation

2.2.4. UART Interface

The ESN7108A contains two UART ports. Port 0 conforms to the industry standard protocol (compatible with 16550A) and contains flow control. Port 1 does not support flow control. Parameters such as data width, number of stop bits, and parity are programmable by the user through the UART interface registers.

2.2.5. GPIO Interface

The eight general-purpose I/O (GPIO) pins on the Cypher ESN7108A allow easy integration with other devices.

The GPIO interface contains the following features:

- Each GPIO pin can be configured as an input or output. If the output mode is not set, the related GPIO pin will drive Hi-Z.
- The rising or falling edge of a GPIO pin can be programmed to generate an interrupt.
- In addition to general GPIO functionality, the GPIO[3:0] pins also support a pulse width modulated (PWM) output mode. If enabled, the GPIO[3:0] bits drive a pulse width modulated wave form. The ratio of the total high level to low level is programmable through the GPIO PWM register. This function is useful to generate firmware controlled static analog voltages to control the front panels back lights, etc.

When a process wants to drive a GPIO bit, it does so by performing a read-modify-write operation. However, if another process is doing the same operation on another bit, a consistency error may occur. The write enable field is provided to mask off the unwanted bit from writing to the GPIO register. The corresponding write enable bits are available on all the GPIO control registers.

2.2.6. I^{2}C Interface

A 2-wire interface is used to connect sensors or other devices using the I^{2}C protocol. The I^{2}C interface is implemented in hardware.

2.2.7. EMI Interface

The External Memory Interface (EMI) connects Flash memory devices to the ESN7108A. The Flash device is read during the boot up stage. The data from the Flash is used for internal programming. The ESN7108A provides support for Flash write cycles over the EMI interface, as well for firmware upgrade and data storage.

The EMI interface controller maps internal 32/16/8 bit accesses onto the extended memory interface, which is 8 bit.

The EMI interface communicates with the external Flash device via:

- 24-bit address bus
- 8-bit data bus
- 4-bit chip select bus
- control and handshake signals

The EMI interface provides a 64 Mbyte address space divided into four 16 Mbyte spaces, one for each of the four Flash devices. The ESN7108A supports connecting to between 1 and 4 devices.

2.2.7.1. EMI Memory Space Mapping

The size of the EMI space is 64MB that is partitioned into four 16 MB sections as shown in table below.

Table 2–4: EMI Memory Space Mapping

Start Address	End Address	Size	Chip Select
1C00_0000	1CFF_FFFF	16MB	CS3
1D00_0000	1DFF_FFFF	16MB	CS2
1E00_0000	1EFF_FFFF	16MB	CS1
1F00_0000	1FFF_FFFF	16MB	CS0

2.2.7.2. EMI Interface Configurable Parameters

The EMI interface provides the following configurable parameters:

- Segment enable/disable
- 5-stage timing (address, setup/CS, strobe, hold, recovery) setting for each stage. Refer to Section 3.2.3., "EMI Interface".
- Intel Cycle type

2.2.7.3. Chip Select 0 Alternate Function

In Self test / Diagnostic mode, segment CS0 is mapped to internal ROM. The POST (Power On Self Test) code starts executing from address 0x1FC0_0000 and performs a simple self test.

After the self test is passed, the processor detects the operating mode. If the operating mode is PCI boot, the processor jumps to an entry point in dram memory and boots the application image. Otherwise, it disables alternate function and maps external flash to 0x1FC0_0000 and continues the boot.

2.2.8. Ethernet MAC (EMAC) Interface

The Ethernet Media Access Controller supports 802.3 Ethernet II packet I/O and interfaces to standard MII PHY chipsets. The ethernet controller is responsible for ingress/egress of 802.3 Ethernet frame data to system memory and is controlled exclusively by the MIPS CPU.

The Ethernet MAC (EMAC) interface contains a dedicated DMA controller and is used for communicating with an external 10/100 PHY. The EMAC contains dedicated transmit and receive channels. The controller contains four internal FIFO’s that allow data to be moved to and from memory.

2.2.9. Audio Input Interface

The Audio Input Interface supports I^{2}S audio input in master or slave mode. The ESN7108A supports an integrated A/V synchronization.

2.2.10.I^{2}S Audio Input

The Cypher ESN7108A supports three standard data formats for I^{2}S audio interface: Right Justified, I^{2}S-Justified and Left Justified. The audio data word length ranges from 8 bits to 24 bits.

Additionally, a special Oki interface protocol is supported, known as Oki mode. This mode is used by Oki voice Codecs like the ML7041. Micronas recommends using this interface to transmit PCM μ-law or PCM A-law 8-bit audio data with long frame synchronous timing.

2.2.10.1.Clock Generator

For master mode I^{2}S/TDM/Oki input, the clock generator must be enabled to generate clock output signals from the reference clock input. By configuring the clock generator, the user can derive the clock output for different sampling rates from the audio reference clock. The clock input frequency can be 24.576 MHz, 22.5792 MHz or 2 MHz.

2.2.10.2.Modes of Operation

The audio controller supports three basic modes of operation as described in table below. Each of these modes is selectable through registers.

Fig. 2–20: Audio Modes of Operation

Mode	Type
I^{2}S	2-channel (left/right)
TDM	6-channel
Oki	1-channel (voice)

I^{2}S Mode

In I^{2}S mode, the audio signal is input as left and right channels. The number of bit clocks per channel depends on the ratio of the clock frequencies.

TDM Mode

In TDM mode, up to 6 channels of audio signal are supported. The number of bit clocks per channel depends on the ratio of the clock frequencies. The bit clock can be a multiple of 44.1 KHz. The relationship between the bit clock and the left and right channel audio signal is shown in Figure 2–21.

As shown in Figure 2–21, within a given channel, the TDM data can be left-, right-, or center-justified relative to the bit clock. For example, if the bit clock is 32-clocks per channel, and each channel is a 16-bit audio sample, the 16-bits can be placed within the 32-bit clocks as shown Figure 2–21.

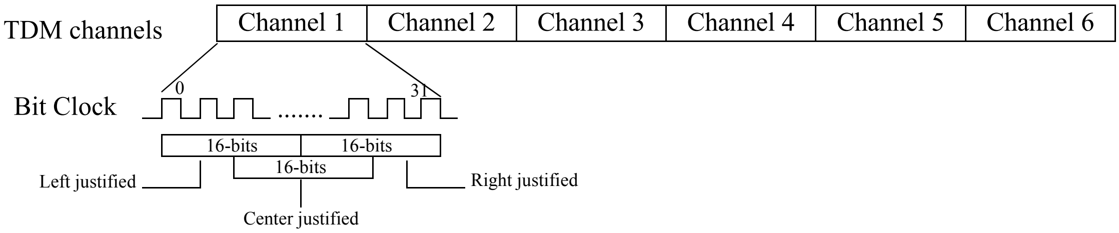


Fig. 2–21: Relationship of Bit Clock to Audio Signal in TDM Mode

Oki Mode

In Oki mode, the audio signal is represented by a single mono channel. The number of bit clocks per channel depends on the ratio of the clock frequencies. The bit clock can be a multiple of 44.1 KHz. The relationship between the bit clock and the single channel audio signal is shown in Figure 2–22.

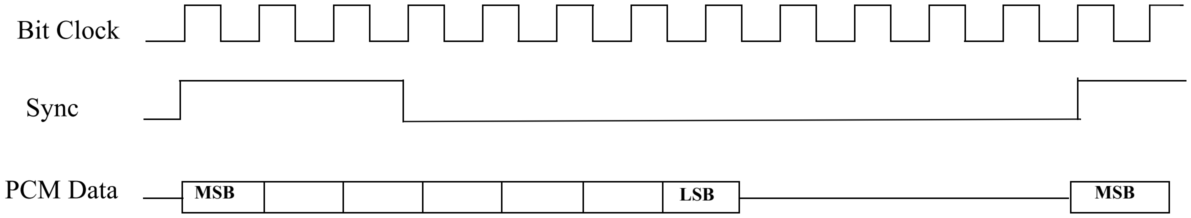


Fig. 2–22: Relationship of Bit Clock to Sync Signal in Oki Mode (long frame sync)

2.2.10.3.Transferring Audio Signals to Memory

The audio interface supports sample data width of 8 - 24 bits/sample. Audio data is transferred via DMA as channels 1 - 6 and is repeated accordingly. For example, if two channels are present (I^2S mode), data is written to the memory as shown inFigure 2–23.

Audio port input signals are sent to the Peripheral DMA (PeriDMA) controller.

Location x	Channel 1
	Channel 2
	Channel 1
	Channel 2
	Channel 1
	Channel 2
	Channel 1
Location x + 7	Channel 2

Fig. 2–23: Storing a Two Channel Audio Signal to Memory

2.2.10.4.Audio/Video Synchronization

The audio block runs off of the bit clock and the VIP module runs off of the 27 MHz video clock. The VIP estimates the amount of time per video frame. This value is converted into the audio clock domain time. In this case two counters are programmed: one counter stores the video frame time, and the other counter stores the error or skew threshold between frames.

2.2.10.5.Master/Slave Operation

The audio interface can operate in either Master or Slave mode. The main difference is that, in master mode, the ESN7108A generates the bit clock and the left/right clock. In slave mode, these clocks are generated by a peripheral audio device that is connected to the audio input port.

2.2.10.6.Audio Byte Swapping

The Cypher ESN7108A allows for byte swapping in 16-bit sampling mode. In this mode the upper and lower bytes of data are swapped such that the lower byte of the audio signal is on the upper byte of data, and the upper byte of audio signal is on the lower byte of data.

2.2.11.HPI Interface

The HPI interface provide a parallel host interface with separate address and data. The host on the HPI interface can communicate with Cypher ESN7108A chip through this interface. By different address, the data read and write are directed to one of the four FIFOs – upload data, upload command, download data and download command FIFOs. The data FIFO have their own DMA channels to DDR SDRAM, and they can be used for streaming data transfer. The command FIFO will generate interrupt internal to Cypher processor, which will be communicating with external HPI host through the upload/download command FIFOs.

2.2.11.1.HPI External Bus Access

The HPI interface operates in asynchronous mode and is configured with a 16-bit data bus width. The HPI interface is configured for an Intel-style interface. Refer to Section 3.2.2., "HPI Interface" for timing information.

2.2.11.2.HPI Configuration and Interrupt

Through the HPI interface, the external Host can perform the following functions:

- Read and write the FIFO
- Access the FIFO status and control registers
- Generate an interrupt

2.2.11.3.FIFO Interface

There are four types of FIFO implemented in HPI block:

1. Download command FIFO: This 16x16 bit FIFO is used to buffer the commands from the HPI Host processor. As long as this FIFO is not full, the HPI host can continue to transfer commands. All commands in the FIFO will be transferred to the internal MIPS processor via the interrupt mechanism.
2. Upload command FIFO: This 16x16 bit FIFO is used to pass the information from the internal processor to the HPI Host. When this FIFO is not empty, an interrupt is issued to the HPI Host.
3. Download data FIFO: This 16x16 bit FIFO is used to download large amounts of data into the ESN7108A. A separate interrupt is generated and sent to the HPI Host when this FIFO is almost empty. Inside the ESN7108A, the DMA engine reads the data from the FIFO and writes it to DRAM.
4. Upload data FIFO: This 512x16 bit FIFO is used to pass data from the internal processor to the HPI Host. The FIFO generates an interrupt to the HPI host when this FIFO is almost full. Inside the ESN7108A, the DMA engine is filling the FIFO constantly.

FIFO Status and Control Registers

The FIFO Status register provides the current word count of the upload FIFO and space count of the download FIFO. Through the FIFO control register, the HPI can set the threshold of the FIFO to generate the interrupt.

Interrupt Status and Control Registers

Internally, the ESN7108A generates two internal interrupts, one based on the Command FIFO status and one based on the Data FIFO status. The Interrupt Control register is used to mask these FIFO’s from interrupting the HPI Host. The Interrupt Status register indicates to the HPI Host which FIFO is generating the interrupt.

2.2.11.4.HPI Configuration and Interrupt

Through the HPI interface, the external Host can perform the following functions:

- Access the power management registers
- Read and write the FIFO
- Access the FIFO status and control registers

– Generate an interrupt

2.2.12.ATAPI Interface

The Cypher ESN7108A supports the industry-standard ATAPI interface for communication with the hard disk drive (HDD). The on-chip ATAPI controller contains the following features:

- Supports programmable I/O (PIO), multi-word DMA, and Ultra ATA 33/66 modes of operation.
- Supports up to Mode 4 timings in PIO mode.
- Supports up to Mode 2 timings in multi-word DMA mode.
- Supports up to Mode 4 timings in Ultra DMA mode.
- Speed-select feature allows timing parameters to be reprogrammed to support any ATA timing mode at any clock frequency.

The on-chip controller provides an interface between the ESN7108A and the IDE/ATA HDD interface. The controller supports multi-word DMA and Ultra DMA transfers between an external IDE/ATA device and the system memory. The controller manages the generation and timing of address, data, and control signals on the IDE interface. Refer to ATAPI standards for Electric characteristics.

2.2.12.1.Modes of Operation

The ESN7108A ATA interface supports three main configurations: PIO, Multi-word DMA, and Ultra DMA. Each of these configurations supports a variety of speed grades. The speed modes and associated clock periods are listed in the table below.

Table 2–5: Speed Modes and Clock Periods

Configuration	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Units
PIO	600	383	240	180	120	ns
Multi-Word DMA	480	150	120	---	---	ns
Ultra DMA	240	160	120	90	60	ns

2.2.12.2.Dedicated DMA Engine

A dedicated DMA engine is used to control data transfers to and from the system memory and the FIFO internal to the ATA controller in the ESN7108A.

2.2.12.3.ATA Interface Interrupts

The ATA controller uses an interrupt-driven protocol and generates an active high interrupt under any of the following conditions.

- For multi-word or Ultra DMA (UDMA) transfers, an interrupt is asserted once the DMA transfer has been completed. For DMA reads (data flows from the external device to the system memory), and interrupt is generated once the last word of data has been written to system memory. For DMA writes (data flows from the system memory to the external device), and interrupt is generated once the last word of data has been written to the external ATA device. Interrupts are cleared by software.
- For PIO data transfers, the interrupt signal is asserted once the ATA device asserts its INTRQ signal. The generation of this signal by the external device occurs after each sector is transferred (read/write sector command), or after the entire transfer has completed (read/write multiple commands).
- When the internal IORDY timer times out. This occurs when the target device does not assert its IORDY signal within a specified period of time.

2.2.12.4.Interface Clocking

The ATA interface supports 33 and 66 MHz clock rates. The default speed is 33 MHz.

2.2.13.Transport Stream Output (TSO)

TSO (Transport Stream Output) handles the PCR correction and provides a FIFO interface for transport stream output. The FIFO interface supports DVB-SPI or DVB-ASI working modes (BS EN 50083-9). Both 188 byte and 204 byte packet lengths are supported.

The MIPS processor performs multiplexing of video and audio and encapsulate the data into 188- or 204-

byte MPEG2 transport stream packets. The MIPS processor inserts the PCR into the packet every N packets. The value of N can be calculated as follows:

$$N= TxCk \times \frac{T}{PLEN}$$

Where:
TxCk = output in bytes per second
T = PCR arrival interval (T ≤40 ms for DVB and ≤100ms for ATSC)
PLEN = packet length of transport stream (188 or 204)

2.2.13.1.Design For Test

The Cypher ESN7108A contains a 5-pin Test Access Port (TAP) interface compliant with IEEE1149.1 JTAG standard for in-system testing.

Micronas implements Multiplexer-based full scan methodologies to achieve high test coverage for stuck-at faults, and at-speed transition and path delay faults for the Device Under Test (DUT).

In addition to DUT, Micronas implements Memory Built In Self Test (MBIST) and marching algorithms to test all RAMs and Multiple Instruction Signature Register (MISR) and data compressor to test the ROM. The MBIST and ROMBIST operations are controlled by TAP controller, which also supports Iddq current measurements.

2.3. High Performance Interconnect

The high performance interconnect in Figure 1–1 provides for the interconnection of each unit in the Micronas. This high performance interconnect performs the following functions:

– Address decoding of master requests

– Routing of requests to the appropriate slaves

– Provide data throttling where required

– Provide arbitration and response in case of simultaneous access of a slave by multiple masters

2.3.1. High Performance Interconnect Memory Map

The various devices are mapped to the high performance interconnect as shown in the Table 2–6.

Table 2–6: Memory Map

Address Range	Definition
0x00000000 - 0x0FFFFFFF	SDRAM space (256 MB)
0x10000000 - 0x17FFFFFF	Unmapped PCI space (128 MB)
0x18000000 - 0x18FFFFFF	Peripheral register space (16 MB)
0x19000000 - 0x19FFFFFF	Reserved space (16 MB)
0x1A000000 - 0x1AFFFFFF	Encoder register space (16 MB)
0x1B000000 -0x1B0FFFFF	DDR register PICA space (1 MB)
0x1B100000 - 0x1B1FFFFF	Error space (1MB)
0x1B200000 - 0x1B2FFFFF	Reserved space (1MB)
0x1B300000 - 0x1B3FFFFF	USB register space (1MB)
0x1B400000 - 0x1B4FFFFF	ATAPI register space (1MB)
0x1B500000 - 0x1B5FFFFF	Color Conversion PICB space (1MB)
0x1B600000 - 0x1BFFFFFF	Error space (10MB)
0x1C000000 - 0x1FFFFFFF	Peripheral Unit boot space (64MB)

Table 2–6: Memory Map, continued

Address Range	Definition
0x20000000 - 0x2FFFFFFF	Reserved space (256MB)
0x30000000 - 0x3FFFFFFF	Error space (256MB)
0x40000000 - 0x7FFFFFFF	Mapped PCI space (1GB)
0x80000000 - 0xFFFFFFFF	Error space (2 GB)

2.4. DDR Memory Controller

The DDR controller provides an interface between the various requesting devices connected to the high performance interconnect, and the DDR SDRAM. The controller provides arbitration functions for SDRAM access and the appropriate refresh control.

The DDR controller performs the following functions:

- Address validation of the various master ports
- Control and data clocking from the external DDR SDRAM
- Management of command, read and write data FIFOs
- Byte masking

The ESN7108A requires 16-bit DDR SDRAM to be used.

2.5. Copy Engine

The Copy Engine copies data from one DDR memory location to another.

2.6. AES Encryption

The AES encryption/decryption block reads data from one DDR memory location, performs AES encryption or decryption, and writes the data to another memory location. This function is only available on the Cypher ESN7108A device. Encryption is not supported on the 7118.

The following algorithms are supported:

- CBC (Cipher Block Chaining)
- CFB (Cipher Feedback)
- CTR (Counterer)
- ECB (Electronic CodeBook)
- OFB (Output Feedback)

2.7. MIPS Processor Interface Unit

The MIPS processor operates at 166 MHz. The core implements the MIPS32™ Instruction Set Architecture (ISA) and contains a 16 dual-entry joint TLB (JTLB) with variable page sizes and 24 Kbyte 3-way set associative instruction and data caches.

2.7.1. EJTAG

The Enhanced JTAG (EJTAG) block allows for single-stepping of the processor as well as instruction and data virtual address/value breakpoints. For more information, see Section 2.9. "JTAG Support" on page 35.

2.8. USB Controller

The USB Controller in the ESN7108A is a serial communications device that conforms to the Universal Serial Bus (USB) Specification 2.0 protocol. The USB Controller acts as a bridge between the USB interface and the high performance interconnect bus. The USB port acts as a master when interfacing to the DDR memory controller. The USB port is in slave mode during communication with either of the MIPS processors.

The USB controller in the ESN7108A also supports the On-The-Go (OTG) specification, providing a low-cost connectivity solution for consumer portable devices such as mobile phones, PDAs, digital still cameras and MP3 players.

When used as a peripheral, IN transactions are processed through Tx endpoints and OUT transactions are processed through Rx endpoints. When used as a host, IN transactions are processed through Rx endpoints and OUT transactions are processed through Tx endpoints. These additional endpoints can be individually configured in software to handle either Bulk transfers (which also allows them to handle Interrupt transfers) or Isochronous transfers.

Each endpoint requires an associated FIFO. The USB controller in the ESN7108A contains a memory interface for connecting to a single block of synchronous single-port RAM which is used for all the endpoint FIFOs.

The FIFO for Endpoint 0 is required to be 64 bytes deep and will buffer 1 packet. The memory interface is configured with regard to the other endpoint FIFOs, which may be from 8 to 8192 bytes in size and can buffer either 1 or 2 packets. Separate FIFOs are associated with each endpoint.

The USB controller provides all of the encoding, decoding and checking needed in sending and receiving USB packets, interrupting the CPU only when endpoint data has been successfully transferred.

When acting as the host for point-to-point communications, the USB controller additionally maintains a frame counter and automatically schedules SOF, Isochronous, Interrupt and Bulk transfers. It also includes support for the Session Request and the Host Negotiation Protocols used in point-to-point communications, details of which are given in the USB On-The-Go supplement to the USB 2.0 specification.

2.8.1. Modes of Operation

The USB controller has two main modes of operation: Peripheral mode and Host mode.

In Peripheral mode, the USB controller encodes, decodes, checks and directs all USB packets sent and received. IN transactions are handled through the Tx FIFOs, while OUT transactions are handled through its Rx FIFOs. This is the mode in which the USB controller needs to be operating when it is acting as the peripheral to a standard USB Host, or as the peripheral in point-to-point communications. Control, Bulk, Isochronous and Interrupt transactions are supported.

In Host mode, the USB controller drives and, within the limits imposed from the USB PHY, powers the USB bus hosting point-to-point communications with a single directly-connected USB function. The USB controller and the USB PHY encode, decode and check the integrity of Control, Bulk, Isochronous and Interrupt OUT and IN transactions scheduled from the software device driver on an available endpoints; the transactions are executed using Tx and Rx queues (FIFOs) whose size is dynamically configured for each endpoint.

Whether initially operates in Host mode or in Peripheral mode depends on whether it is being used in an 'A' device or a 'B' device, which in turn depends on whether the input is low or high. When the USB controller is operating as an 'A' device, it is initially configured to operate in Host mode. When operating as a 'B' device, the USB controller is initially configured to operate in Peripheral mode.

2.8.2. USB Interface Configuration

The USB On-The-Go (OTG) core implemented in the Cypher ESN7108A supports the following features:

– 8-bit ULPI interface

– Nine transmit (Tx) endpoints.

– Seven receive (Rx) endpoints and the control endpoint (EP0).

– 16KB SPRAM to allow for simultaneous access of endpoints and their requirements.

– 8 DMA channels.

– Dynamic FIFO sizing. Firmware has full control over the FIFO address and FIFO size for each indexed endpoint.

– All the endpoints are indexed by the firmware and hence endpoint attributes like type, mps, mult can be selected dynamically. In addition to these attributes, endpoint direction can also be configured in accordance with Tx endpoints to be numbered from 1 - 9 and Rx endpoints to be numbered from 1 - 7.

– Tx and Rx endpoints do not share a FIFO.

– Little endian byte ordering.

– Soft connect/disconnect option.

– Bulk endpoints support multiple packets and ISO endpoints support high-bandwidth transactions.

– Vendor control registers are not included in the Cypher ESN7108A USB core.

2.9. JTAG Support

The JTAG interface consists of 5 pins: JTAG_TDI, JTAG_TCK, JTAG_TMS, JTAG_TDO and JTAG_TRST_N. The TAP complies with IEEE1149.1 JTAG standard with the mandatory instructions of SAMPLE/PRELOAD, EXTEST and BYPASS. In addition Micronas-specific ENTER and EXIT instructions for entering and exiting MIPS EJTAG controller.The TAP controller consists of a 5-bit instruction register that is encoded as shown in table below.

Table 2–7: JTAG Opcodes

TAP Instruction	Opcode[1]	Register	Comments
BYPASS	11111	BYPASS	JTAG mandatory BYPASS instruction.
EXTEST	00000	BOUNDARY	JTAG mandatory EXTEST instruction.
SAMPLE/PRELOAD	00001	BOUNDARY	JTAG mandatory SAMPLE/PRELOAD instruction.
IDCODE	00010	IDCODE	JTAG optional IDCODE instruction.
ENTER EJTAG	00101		Micronas specific instruction. Used to enter MIPS EJTAG mode.
EXIT EJTAG	00100		Micronas specific instruction. Used to exit MIPS EJTAG mode.

[1] All values not shown are reserved.

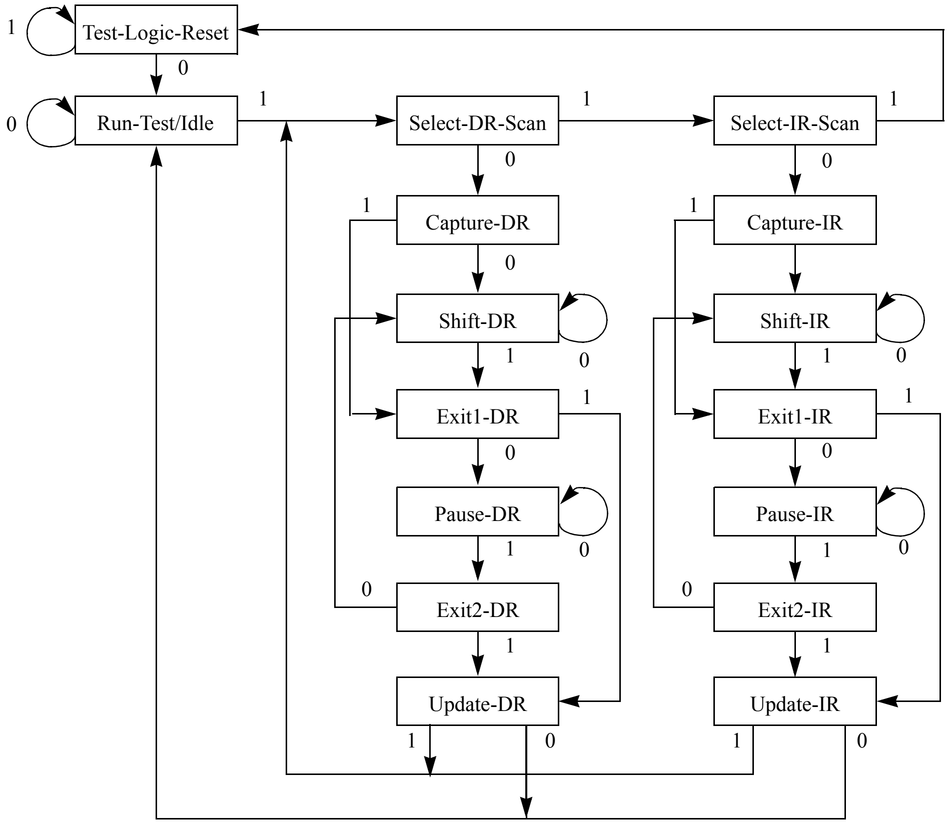


Fig. 2–24: TAP Controller State Diagram

2.9.1. BYPASS

The bypass instruction allows a single bit register (the BYPASS register) be selected between JTAG_TDI and JTAG_TDO such that test operation on the device is bypassed. In a daisy-chained boundary scan architecture, this mode allows devices unintended for test to be bypassed.

This is a public instruction mandatory for IEEE1149.1.

2.9.2. EXTEST

EXTEST allows testing of off-chip circuitry and board level interconnections. Prior to applying EXTEST, a pattern is first shifted into the boundary scan register using the SAMPLE/PRELOAD instruction. When EXTEST is applied, the output pins are used to apply test stimuli, while the input pins are used to capture test results received from an external source or a neighboring device.

This is a public instruction mandatory for IEEE1149.1.

2.9.3. SAMPLE/PRELOAD

The SAMPLE/PRELOAD instruction allows a snapshot of the normal operation of the component to be taken and examined. This instruction should not interfere with the normal operation of the on-chip system logic. SAMPLE allows a snapshot of internal states to be shifted out to JTAG_TDO on each JTAG_TCK clock tick. PRELOAD allows an initial data pattern to be placed at the latched parallel outputs of the boundary-scan register cells.

This is a public instruction mandatory for IEEE1149.1.

2.9.4. IDCODE

The 32-bit IDCODE instruction allows the device identification register to be read serially from the component through the JTAG ports. When the IDCODE instruction is selected, the vendor identification code is loaded into the device identification register. The device identification register is then shifted out through JTAG_TDO for observation.

The 32-bit device identification register is configured as follows:

Version (MSB)	0000
Part Number	depends on bound-out option
AES_EN tied low (0)	0111 0001 0001 1000
AES_EN tied high (1)	0111 0001 0000 1000
Vendor ID	0101 110 0 011
Mandatory (LSB)	1 (IEEE mandatory for bit [0])

This is a public instruction that is optional for IEEE1149.1.

2.9.5. ENTER EJTAG

The JTAG ports are shared with both JTAG and MIPS EJTAG. When this instruction is issued, the JTAG port is connected to MIPS EJTAG for MIPS debugging.

This is a Micronas-specific instruction.

2.9.6. EXIT EJTAG

This instruction exists MIPS EJTAG mode and resumes TAP controller to normal boundary scan (JTAG) mode.

This is a Micronas-specific instruction.

3. Specifications

3.1. Electrical Specifications

3.1.1. Absolute Maximum Ratings

Table 3–1: Absolute Maximum Ratings[1]

Parameter	Value
Voltage on CORE Pins Relative to GND	-0.3V to 1.32V
Voltage on VIO33 pins relative to GND	TBD
Voltage on Input or I/O pins relative to GND	TBD
Voltage on VIO25 pins relative to GND	-0.3V to 2.75V
Operating temperature, T_A (Ambient)	0°C to 70°C
Storage temperature	-55°C to +125°C

[1] Stresses greater than those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only. Functional operation of the device at these, or at conditions above those indicated in the operational sections of this specification, is not implied. Exposure to absolute maximum rating conditions for extended periods may affect reliability.

3.1.2. Recommended Operating Conditions

Table 3–2: Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Unit	Note
Core Supply Voltage	V_{CORE}	1.188	1.25	1.313	V	
IO Supply Voltage	V_{IO33}	3.0	3.3	3.6	V	
	V_{IO25}	2.375	2.5	2.625	V	
Core Supply Current	I_{C_CORE}		770		mA	1
IO Supply Current	I_{C_IO33}		TBD		mA	1
	I_{C_IO25}		TBD		mA	
Power Dissipation	P_D		1.1		W	2
Analog and Digital Supply Voltage for PLL	AV_{DD_PLL}, DV_{DD_PLL}	1.188	1.25	1.313	V	
Input High Voltage	V_{IH}		TBD		V	
Input Low Voltage	V_{IL}		TBD		V	
Threshold Voltage	V_T		TBD		V	
Input leakage current @V_I=3.3V or 0V w/ pull-up or pull-down resistor	I_{IP}	-165		+165	µA	3
Low Level Output Current						
I^2S, MII, UART, TSO, EMICS	$I_{OL(1)}$	2.6		4.8	mA	V_{OL}= 0.35V

Table 3–2: Recommended Operating Conditions, continued

Parameter	Symbol	Min	Typ	Max	Unit	Note
USB, HPI, GPIO	$I_{OL(3)}$	7.8		14.4	mA	V_{OL}= 0.35V
EMI (except CS pins)	$I_{OL(4)}$	10.5		19.5	mA	V_{OL}= 0.35V
ATAPI	$I_{OL(6)}$	13.1		24.3	mA	V_{OL}= 0.35V
High Level Output Current						
I^2S, MII, UART, TSO, EMICS	$I_{OH(1)}$	1.8		3.5	mA	V_{OH}=V_{IO33}-0.35V
USB, HPI, GPIO	$I_{OH(3)}$	5.6		10.4	mA	V_{OH}=V_{IO33}-0.35V
EMI (except CS pins)	$I_{OH(4)}$	7.5		13.8	mA	V_{OH}=V_{IO33}-0.35V
ATAPI	$I_{OH(6)}$	11.2		20.8	mA	V_{OH}=V_{IO33}-0.35V

1: Estimated supply current at V_{CORE} = 1.25V, V_{IO25} = 2.5V, V_{IO33} = 3.3V full-D1 @ 30 fps.
2: Estimated power consumption at VCORE = 1.25V, VIO25 = 2.5V, VIO33 = 3.3V full-D1 @ 30 fps.
3: On applicable input pins.

3.2. AC Specifications

3.2.1. Video Input Interface

Table 3–3: Video Input Interface AC Characteristics

Parameter	Symbol	Min	Typ	Max	Unit	Note
PDATA Setup time	t_{DSU}	10	-	-	ns	
PDATA Hold time	t_{DHD}	5	-	-	ns	
Reference Signal Setup time	t_{RSU}	10	-	-	ns	
Reference Signal Hold time	t_{RHD}	5	-	-	ns	
Period of PCLK	t_{PC}	37	-	100	ns	
PCLK Duty factor		45%		55%		
PCLK clock rising transition time	t_{RISE}			2	ns	

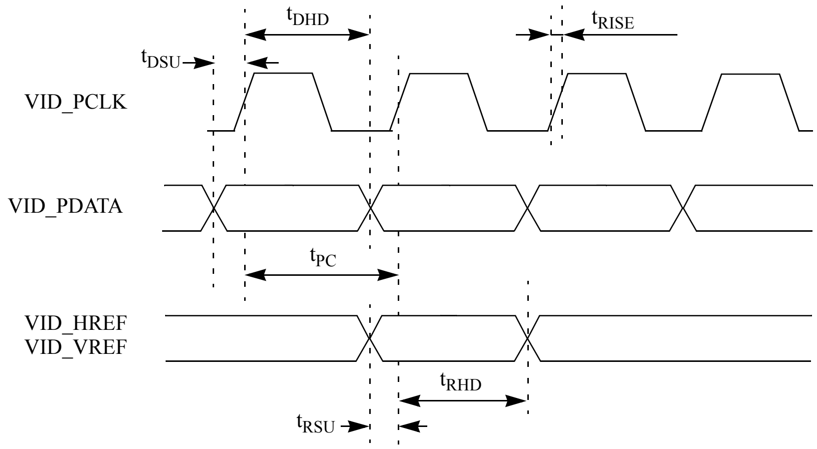


Fig. 3–1: Video Input Interface AC Characteristics

3.2.2. HPI Interface

The HPI interface operates in asynchronous mode and is configured with a 16-bit data bus width, which is set by the boot-up. The HPI interface is configured for an Intel-style interface.

Table 3–4: HPI Interface AC Characteristics

Parameter	Symbol	Min	Typ	Max	Unit	Note
HPI_CS_N low to HPI_RD_N low	t_{CR}	0		100	ns	
HPI_RD_N low to data valid	t_{RD}	24	36	48	ns	
HPI_RD_N low time	t_{RL}	24			ns	
HPI_RD_N high time	t_{RH}	24			ns	
Data hold time from HPI_RD_N high	t_{DZ}	1.5	2	3	ns	[1]
HPI_RD_N high to HPI_CS_N high	t_{RC}	0			ns	
HPI_CS_N low to HPI_WR_N low	t_{CW}	0			ns	
HPI_ADDR valid to HPI_WR_N low	t_{AW}	0			ns	
HPI_WR_N low time	t_{WL}	24	36	48	ns	
HPI_WR_N high time	t_{WH}	24			ns	
HPI_D data valid to HPI_WR_N high	t_{DW}	0			ns	
HPI_D data hold from HPI_WR_N high	t_{DWH}	24	36	48	ns	
HPI_WR_N high to HPI_CS_N high	t_{WC}	0			ns	

[1] Preliminary Numbers

3.2.2.1. 16-Bit Asynchronous Intel-Style Addressing Mode

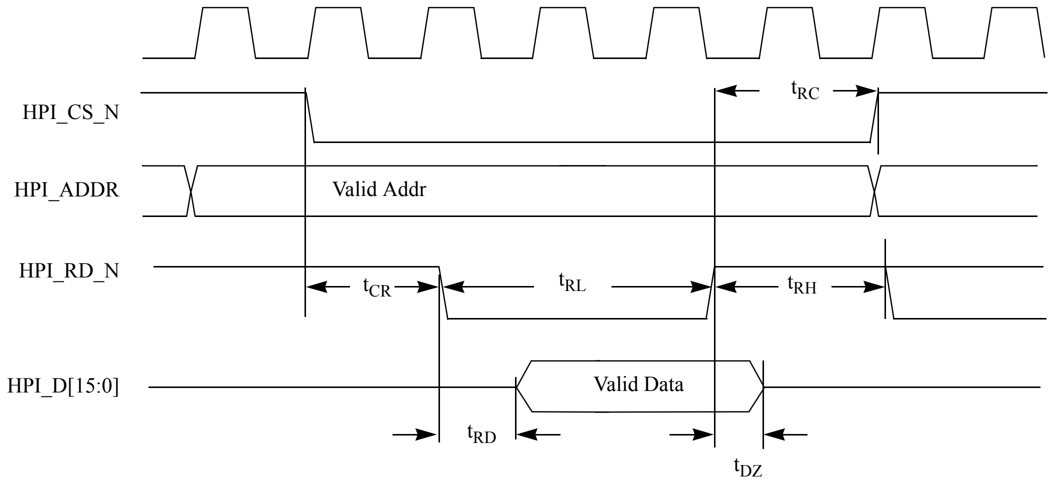


Fig. 3–2: 16-Bit Intel-Style Read Operation

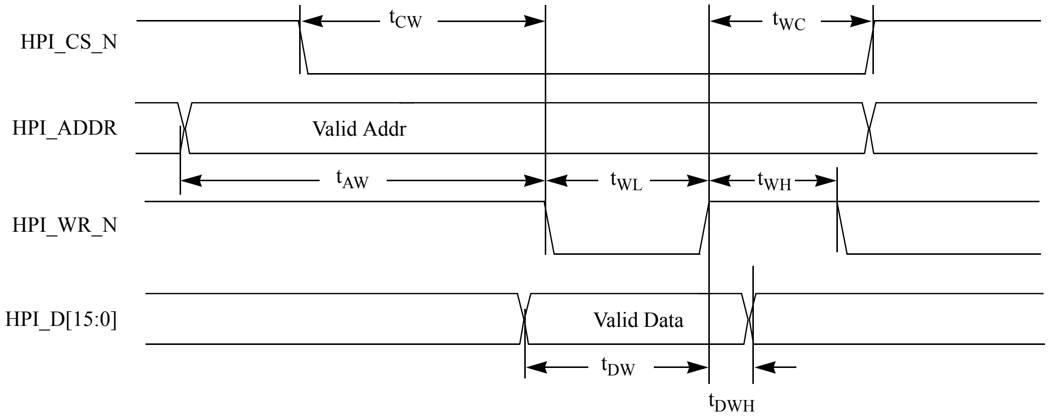


Fig. 3–3: 16-Bit Intel-Style Write Operation

3.2.3. EMI Interface

The EMI interface can be configured to operate in either Intel mode or Motorola mode during power-on reset. Each of these interface timings is described in the table below.

Table 3–5: EMI Interface AC Characteristics

Parameter	Symbol	Min	Typ	Max	Unit	Note
EMI_CS_N low to EMI_RD_N low	t_{CR}	12	36	10,752	ns	
EMI_RD_N low to data valid	t_{RD}	12	36	10,752	ns	
EMI_RD_N low time	t_L	12	578	96,000	ns	
EMI_RD_N high time	t_H	12	30	10,752	ns	
Data hold time from EMI_RD_N high	t_{DZ}	0			ns	
EMI_RD_N high to EMI_CS_N high	t_{RC}	12	578	96,000	ns	
EMI_CS_N low to EMI_WR_N low	t_{CW}	12	36	10,752	ns	
EMI_ADDR to EMI_WR_N setup	t_{AW}	12	36	10,752	ns	
EMI_WR_high to EMI_CS_N high	t_{WC}	12	578	10,752	ns	
EMI_DATA setup to EMI_WR_N high	t_{DW}	12	578	10,752	ns	
EMI_DATA hold to EMI_WR_N high	t_{DHW}	12	578	10,752	ns	

Intel Timing

The Intel mode uses the EMI_CS_N signals to qualify the address and whole access. This mode uses the rising edge of EMI_WR_N to latch write data. Driving EMI_RD_N low enables the device output. Read data is then latched on the rising edge of EMI_RD_N.

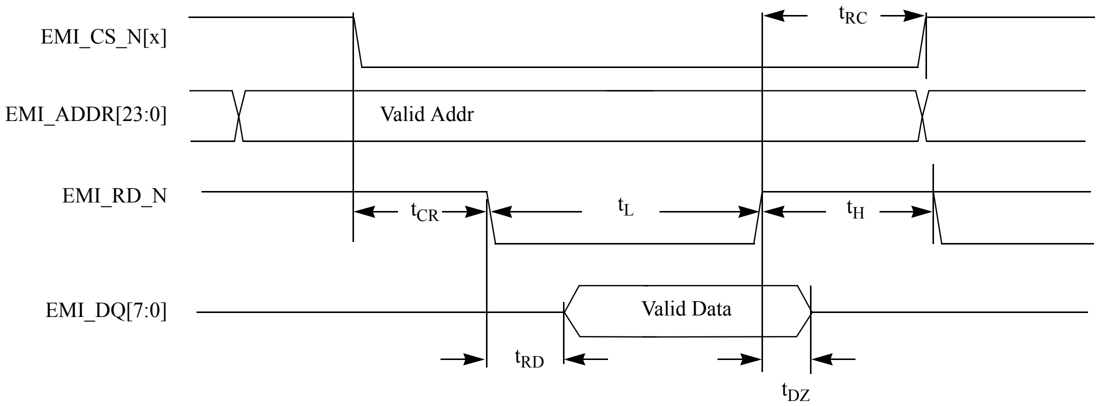


Fig. 3–4: Intel Mode Read Cycle

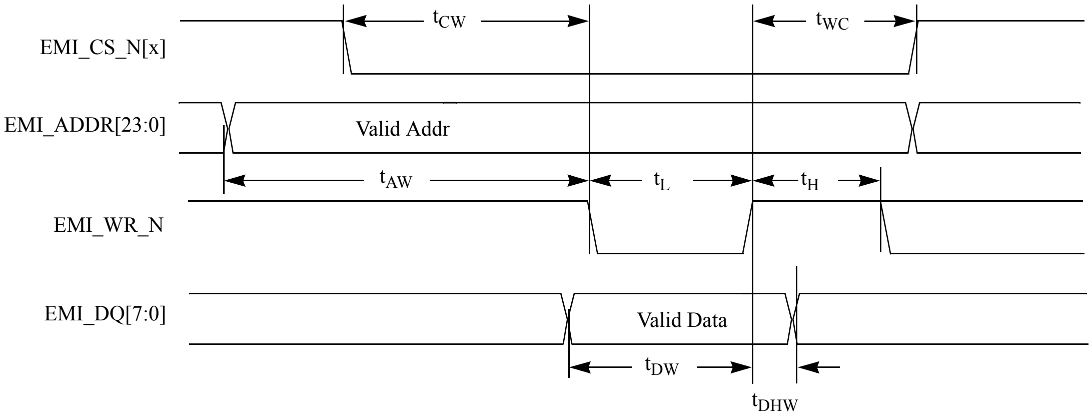


Fig. 3–5: Intel Mode Write Cycle

3.2.4. PCI/Cardbus Interface

For AC specifications, refer to table below.

Table 3–6: PCI AC Specifications

Parameter	Symbol	Condition	Min	Max	Unit	Note
AC Drive Points						
Switching Current High, minimum	$I_{OH (AC, min)}$	$V_{out} = 0.3V_{cc}$	-39.6	-	mA	1
Switching Current High, maximum	$I_{OH (AC, max)}$	$V_{out} = 0.7V_{cc}$	-	-105.6	mA	
Switching Current Low, minimum	$I_{OL (AC, min)}$	$V_{out} = 0.6V_{cc}$	52.8	-	mA	1
Switching Current Low, maximum	$I_{OL (AC, max)}$	$V_{out} = 0.18V_{cc}$	-	125.4	mA	
DC Drive Points						
Output high voltage	V_{OH}	I_{out} = -0.5 mA	2.97		V	2
Output low voltage	V_{OL}	I_{out} = 1.5 mA		0.33	V	2
Slew Rate						
Output rise slew rate	t_r	$0.3V_{cc}$ to $0.6V_{cc}$	1	4	V/ns	3
Output fall slew rate	t_f	$0.6V_{cc}$ to $0.3V_{cc}$	1	4	V/ns	3
Clamp Current						
High clamp current	I_{ch}	$V_{cc}+4>V_{in}>=V_{cc}+1$	$25+(V_{in}-2.33)/0.015$	-	mA	
Low clamp current	I_{cl}	$-3 < V_{in}<=-1$	$-25+(V_{in}+1)/0.015$	-	mA	

1. This specification does not apply to **PCI_CLK** and **PCI_RST_N** which are system outputs. “Switching Current High” specifications are not relevant to **PCI_SERR_N**, **PCI_PME_N**, **PCI_ INTA_N**, **PCI_INTB_N**, **PCI_INTC_N**, and **PCI_INTD_N** which are open drain outputs.
2. These DC values are duplicated and are included here for completeness.
3. This parameter is to be interpreted as the cumulative edge rate across the specified range rather than the instantaneous rate at any point within the transition range. The specified load is optional. The designer may elect to meet this parameter with an unloaded output per revision 2.0 of the PCI Local Bus Specification. However, adherence to both maximum and minimum parameters is required (the maximum is not simply a guideline). Rise slew rate does not apply to open drain outputs.

33MHz Timing Parameters

Table 3–7: 33MHz Timing Parameters

Parameter	Symbol	Min	Max	Unit	Note
PCI_CLK to Signal Valid Delay -bused signals	T_val	2	11	ns	1, 2, 3, 8
PCI_CLK to Signal Valid Delay - point to point signals	T_val(ptp)	2	12	ns	1, 2, 3, 8
Float to Active Delay	T_on	2		ns	1, 7, 8
Active to Float Delay	T_off		28	ns	1, 7
Input Setup Time to **PCI_CLK** - bused signals	T_su	7		ns	3, 4, 8
Input Setup Time to **PCI_CLK** - point to point signals	T_su (ptp)	10, 12		ns	3, 4
Input Hold Time from **PCI_CLK**	T_h	0		ns	4
Reset Active Time after power stable	T_rst	1		ms	
Reset Active Time after **PCI_CLK** stable	T_rst-clk	100		μs	
Reset Active to output float delay	T_rst-off		40	ns	6
PCI_RST_N high to first Configuration access	T_rhfa	2^{25}		clocks	
PCI_RST_N high to first **PCI_FRAME_N** assertion	T_rhff	5		clocks	
CLK Cycle Time	T_cyc	30	•	ns	9
CLK High Time	T_high	11		ns	
CLK Low Time	T_low	11		ns	

1. See the timing measurement conditions. It is important that all driven signal transitions drive to their V_{oh} or V_{ol} level within one T_{cyc}.
2. Minimum times are measured at the package pin with the load circuit. Maximum times are measured with the load circuit.
3. **PCI_REQ_N** and **PCI_GNT_N** are point-to-point signals and have different input setup times than do bused signals. All other signals are bused.
4. See the timing measurement conditions.
5. All output drivers must be floated when **PCI_RST_N** is active.
6. These values are duplicated and are included here for comparison.
7. For purposes of Active/Float timing measurements, the Hi-Z or "off" state is defined to be when the total current delivered through the component pin is less than or equal to the leakage current specification.
8. Setup time applies only when the device is not driving the pin. Devices cannot drive and receive signals at the same time.
9. In general, all PCI components must work with any clock frequency between nominal DC and 33 MHz. Device operational parameters at frequencies under 16 MHz may be guaranteed by design rather than by testing. The clock frequency may be changed at any time during the operation of the system so long as the clock edges remain "clean" (monotonic) and the minimum cycle and high and low times are not violated. For example, the use of spread spectrum techniques to reduce EMI emissions is included in this requirement. The clock may only be stopped in a low state. A variance on this specification is allowed for components designed for use on the system motherboard only. These components may operate at any single fixed frequency up to 33 MHz and may enforce a policy of no frequency changes.

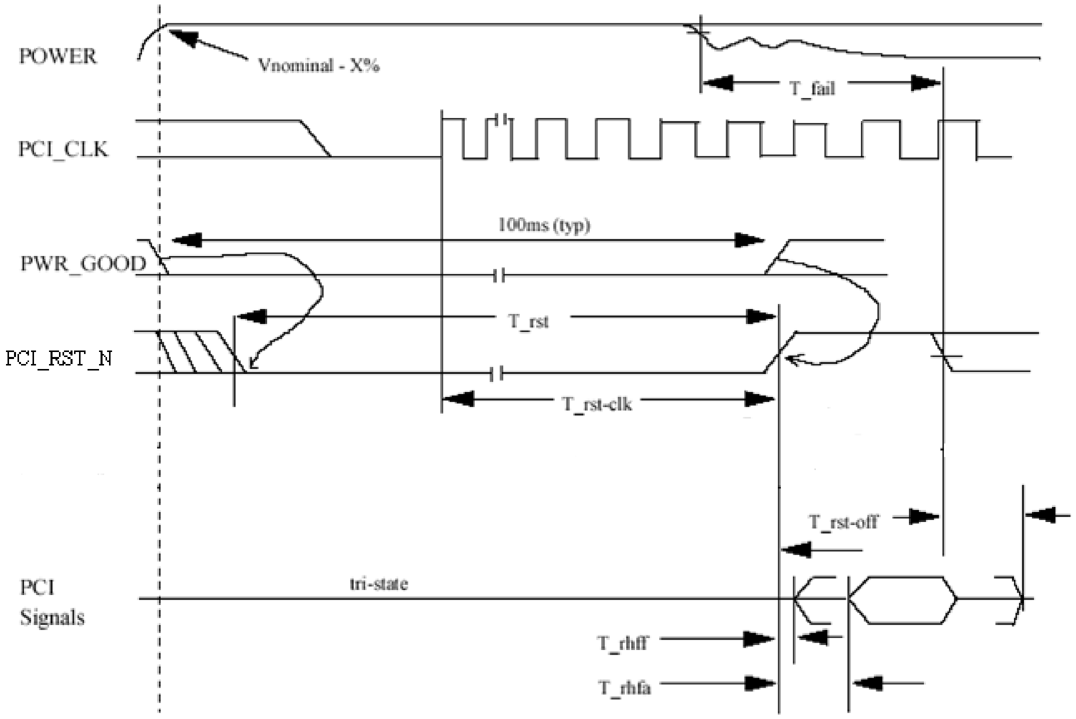


Fig. 3–6: Reset Timing

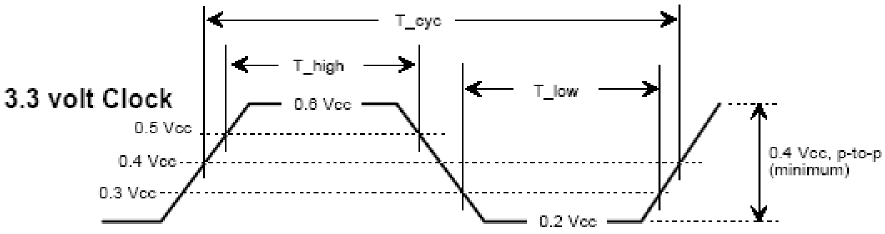


Fig. 3–7: Clock Wave forms

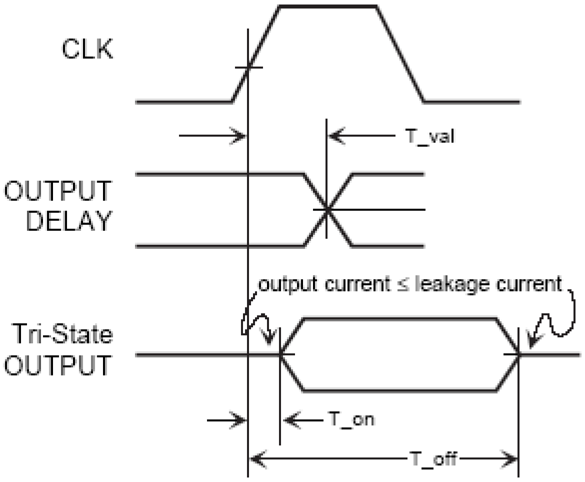


Fig. 3–8: Output Timing Measurement Conditions

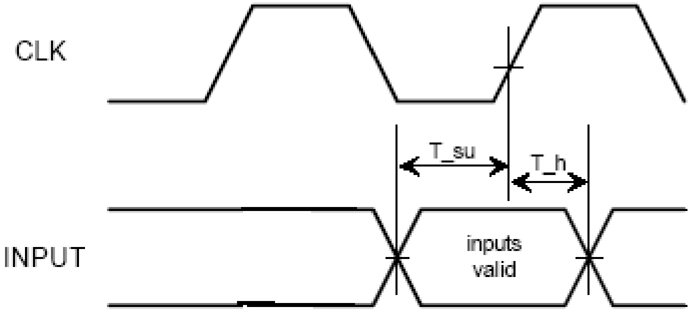


Fig. 3–9: Input Timing Measurement Conditions

3.2.5. I^{2}S Audio Interface

Table 3–8: I^{2}S Interface AC Characteristics

Parameter		Symbol	Min	Typ	Max	Unit	Note
I^{2}S/OKI Slave mode	BCLK cycle	t_{BCLK}	-	326	-	ns	
	SD/LRCLK Setup time	t_{BSU}	5	-	-	ns	
	SD/LRCLK Hold time	t_{BHD}	5	-	-	ns	
I^{2}S Master mode	MCLK cycle	t_{MCLK}	-	18	-	ns	
	BCLK/LRCLK Setup time	t_{MSU}	-	-	TBD	ns	
	BCLK/LRCLK Hold time	t_{MHD}	7	-	-	ns	
	LRCLK Setup time with respect to BCLK	t_{LBSU}	TBD				1
	LRCLK Hold time with respect to BCLK	t_{LBHD}	TBD				

1. By default, LRCLK toggles at falling edge of BCLK by design.

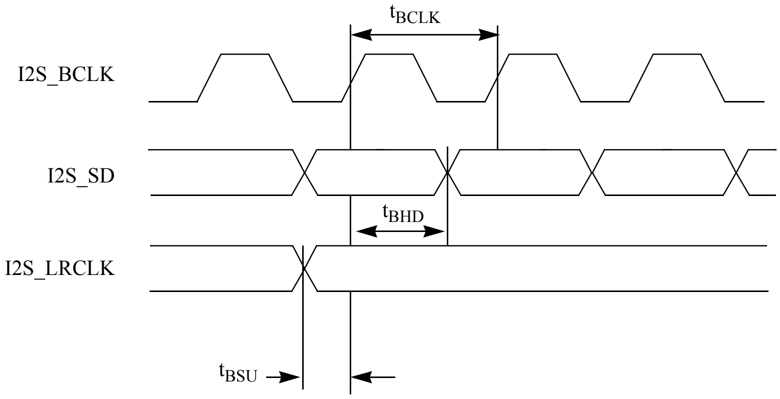
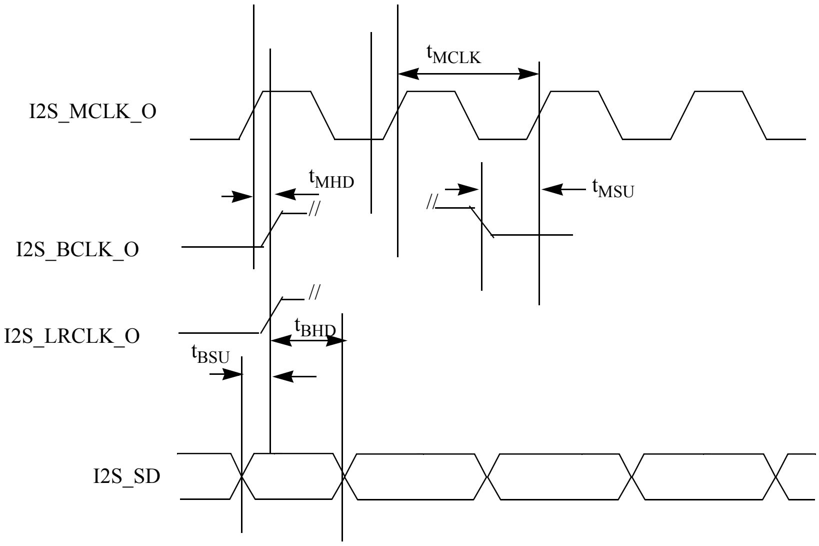
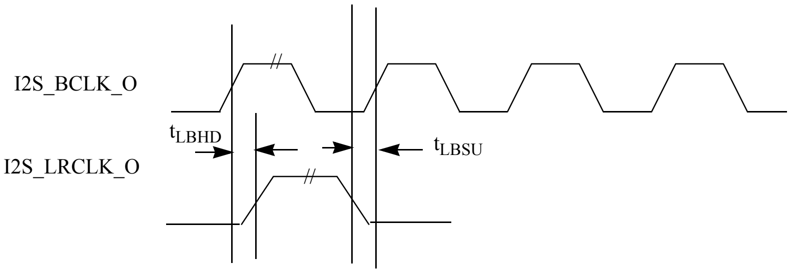


Fig. 3–10: Audio Input Interface Timing - I^{2}S/OKI Slave Mode



(a)



(b)

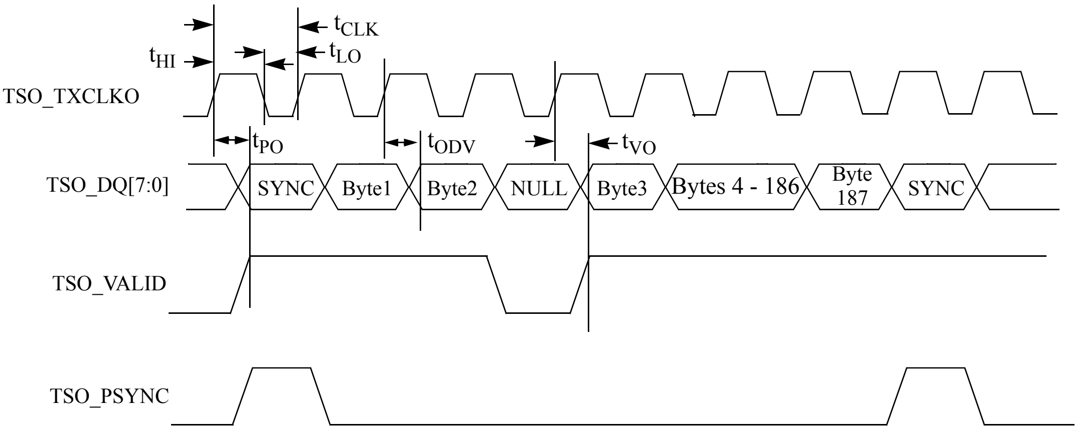
Fig. 3–11: Audio Input Interface Timing - I^{2}S Master Mode

3.2.6. TSO Interface

Table below shows the TSO Interface AC characteristics.

Table 3–9: TSO Interface AC Characteristics

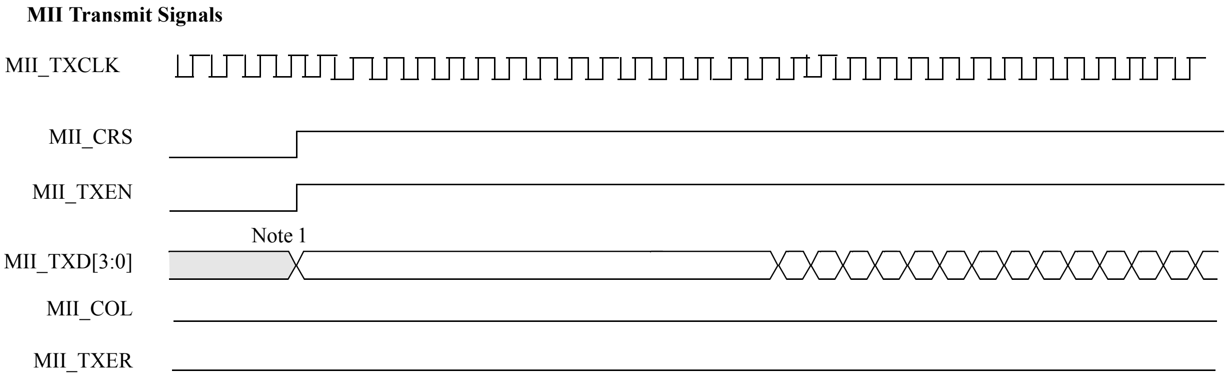
Parameter	Symbol	Min	Typ	Max	Unit	Note
TSO clock high time	t_{HI}	15.15			ns	serial
		15.15			ns	parallel
TSO clock low time	t_{LO}	15.15			ns	serial
		15.15			ns	parallel
TSO clock period	t_{CLK}	30.30			ns	serial
		30.30			ns	parallel
TSO_DQ output data valid time from TSO_TCLK rising edge	t_{ODV}	2.0	TBD	18.3	ns	
TSO_VALID output delay from TSO_TCLK rising edge	t_{VO}	2.0	TBD	18.3	ns	
TSO_PSYNC valid output delay from TSO_TCLK rising edge	t_{PO}	2.0	TBD	18.3	ns	



Note: In mode 0, the NULL should be at least 2 bytes wide.

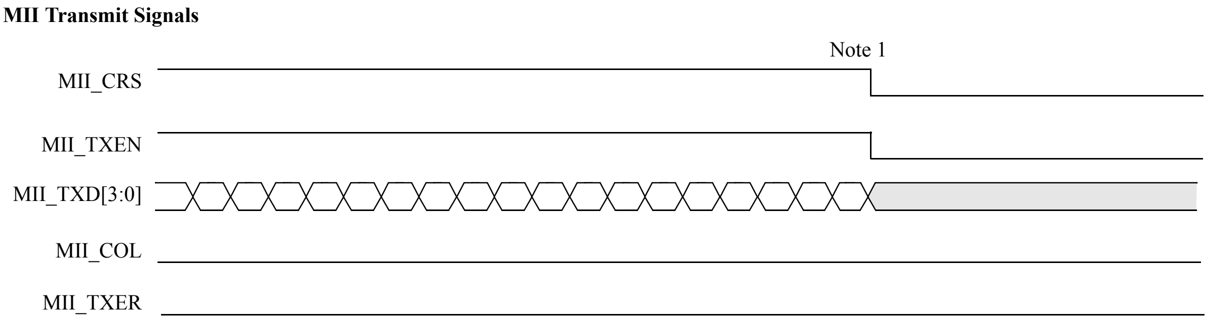
Fig. 3–12: Mode 0/3

3.2.7. Ethernet Interface



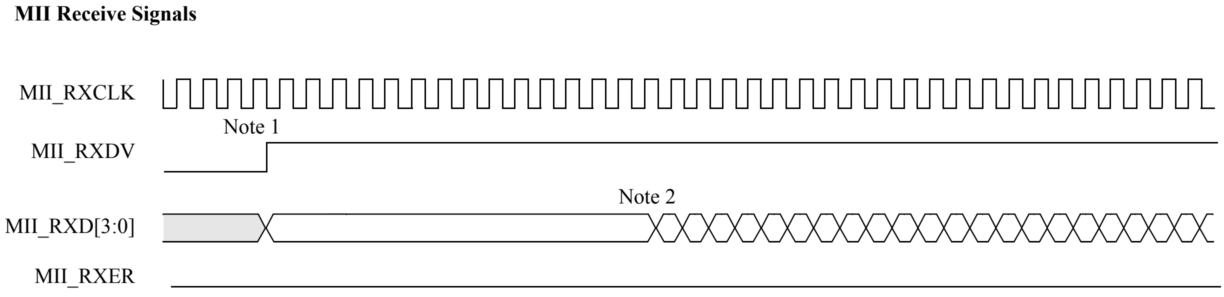
1. The MII begins transmitting the packet preamble and SFD indicated by MII_TXEN assertion, and is followed by the packet’s data.

Fig. 3–13: Start of Transmit



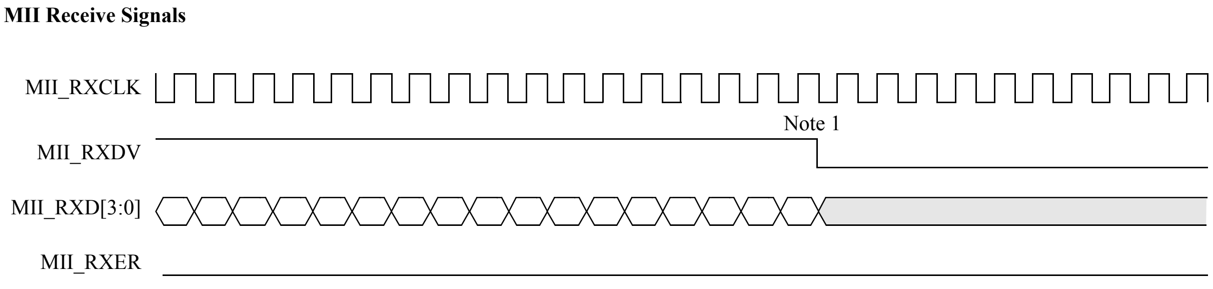
1. After the last nibble of the packet’s FCS is sent out to the wire the MII_TXEN signal is de-asserted.

Fig. 3–14: End of Transmit



1. The MII_RXDV signal is asserted, indicating valid data on the receive bus (MII_RXD[3:0]).
2. The MII looks for the SFD, indicating the start of a packet.

Fig. 3–15: Receive Start of Frame



1. The MII_RXDV signal is de-asserted indicating the end of valid data on the MII_RXD[3:0] bus.

Fig. 3–16: Receive End of Frame

Table 3–10: Timing of Transmission Cycle for 100Mb/s

Parameter	Symbol	Min	Typ	Max	Unit
MII_TXCLK high pulse width	t_1	14		26	ns
MII_TXCLK low pulse width	t_2	14		26	ns
MII_TXCLK duty cycle		35%		65%	
MII_TXCLK period	t_3		40		ns
MII_TXEN, MII_TXD[3:0] setup to MII_TXCLK rising edge	t_4	20			ns
MII_TXEN, MII_TXD[3:0] hold after MII_TXCLK rising edge	t_5	10			ns

Table 3–11: Timing of Transmission Cycle for 10Mb/s

Parameter	Symbol	Min	Typ	Max	Unit
MII_TXCLK high pulse width	t_1	140		260	ns
MII_TXCLK low pulse width	t_2	140		260	ns
MII_TXCLK duty cycle		35%		65%	
MII_TXCLK period	t_3		400		ns
MII_TXEN, MII_TXD[3:0] setup to MII_TXCLK rising edge	t_4	20			ns
MII_TXEN, MII_TXD[3:0] hold after MII_TXCLK rising edge	t_5	10			ns

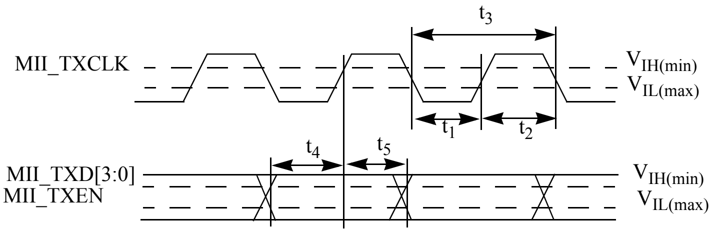


Fig. 3–17: Transmission Cycle Timing

Table 3–12: Timing of Reception Cycle for 100Mb/s

Parameter	Symbol	Min	Typ	Max	Unit
MII_RXCLK high pulse width	t_1	14		26	ns
MII_RXCLK low pulse width	t_2	14		26	ns
MII_RXCLK duty cycle		35%		65%	
MII_RXCLK period	t_3		40		ns
MII_RXEN, MII_RXD[3:0] setup to MII_RXCLK rising edge	t_4	10			ns
MII_RXEN, MII_RXD[3:0] hold after MII_RXCLK rising edge	t_5	10			ns

Table 3–13: Timing of Reception Cycle for 10Mb/s

Parameter	Symbol	Min	Typ	Max	Unit
MII_RXCLK high pulse width	t_1	140		260	ns
MII_RXCLK low pulse width	t_2	140		260	ns
MII_RXCLK duty cycle		35%		65%	
MII_RXCLK period	t_3		400		ns
MII_RXEN, MII_RXD[3:0] setup to MII_RXCLK rising edge	t_4	10			ns
MII_RXEN, MII_RXD[3:0] hold after MII_RXCLK rising edge	t_5	10			ns

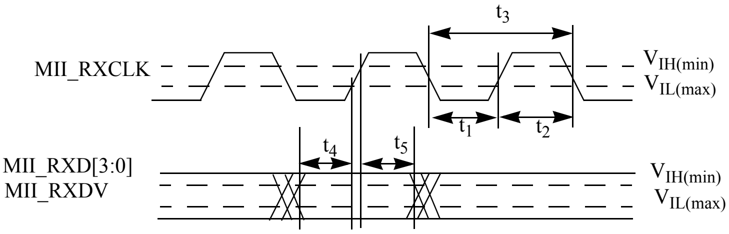


Fig. 3–18: Reception Cycle Timing

3.2.8. MDIO Interface

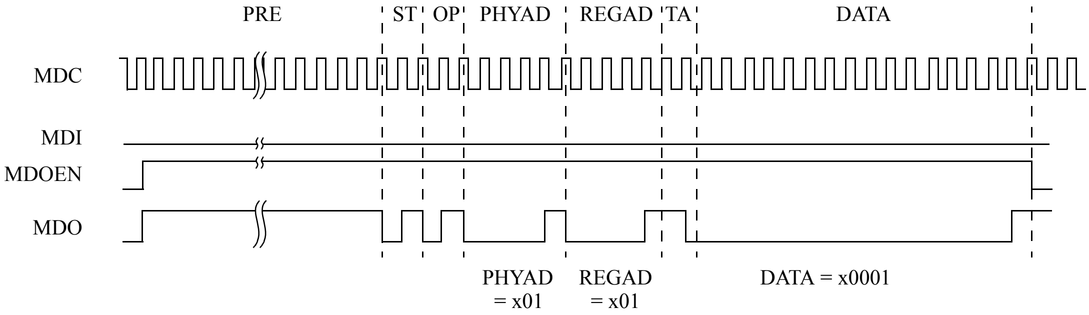


Fig. 3–19: MII Management Write Sequence

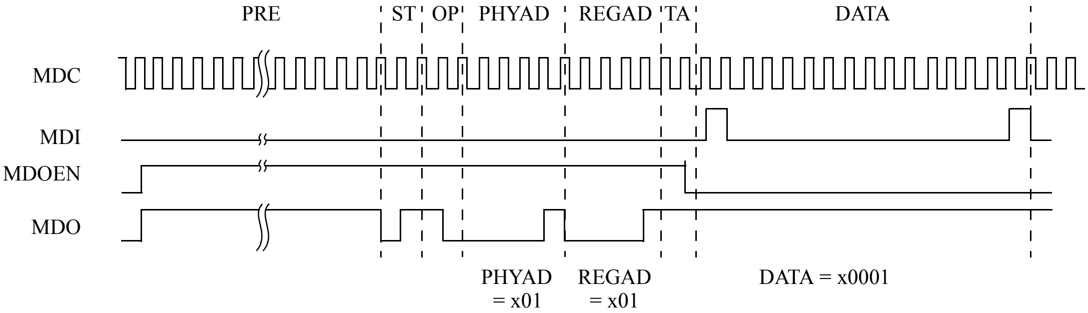


Fig. 3–20: MII Management Read Sequence

Table 3–14: MDIO Interface Timing

Parameter	Symbol	Min	Typ	Max	Unit
MDC high pulse width	t_1	84			ns
MDC low pulse width	t_2	84			ns
MDC period	t_3	24		168	ns
MDIO setup to MDC rising edge	t_4	10			ns

Table 3–14: MDIO Interface Timing, continued

Parameter	Symbol	Min	Typ	Max	Unit
MDIO hold time from MDC rising edge	t_5	10			ns
MDIO valid from MDC rising edge	t_6	0		300	ns

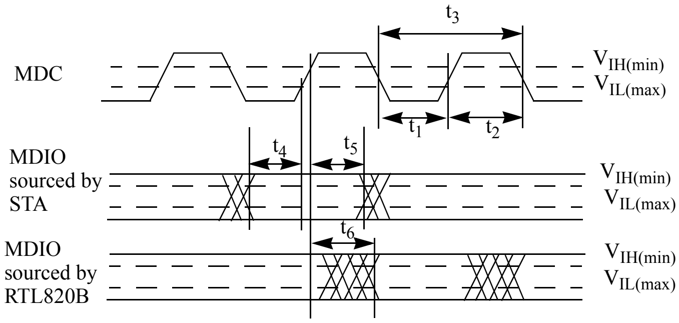


Fig. 3–21: MDIO Interface timings

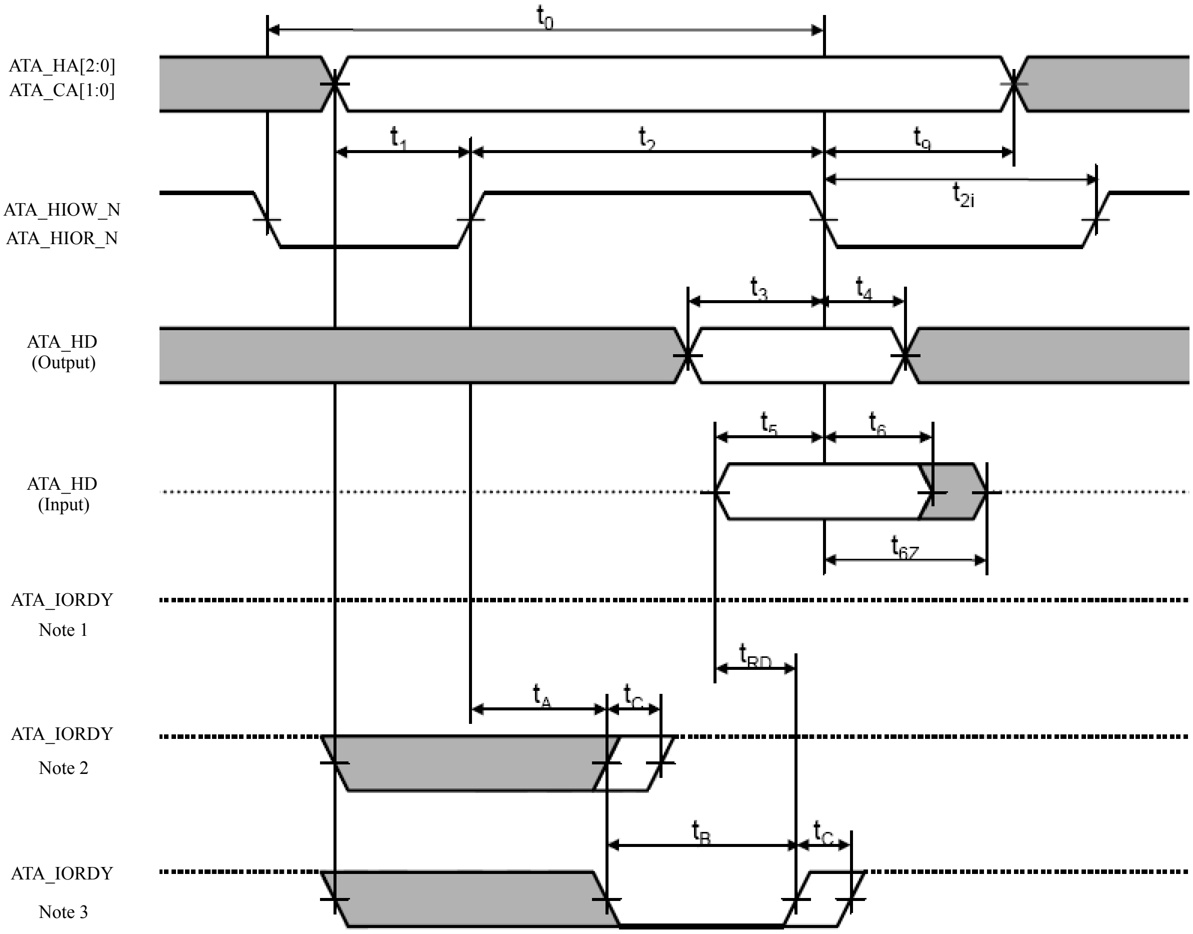
3.2.9. ATAPI HDD Interface

IDE Interface

Table 3–15: IDE Interface PIO Data Transfer Timing Parameter

PIO Timing Parameters (in ns)			Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Note
t_0	Cycle time	(min)	600	383	240	180	120	1
t_1	Address valid to ATA_HIOR_N/ ATA_HIOW_N setup	(min)	70	50	30	30	25	
t_2	ATA_HIOR_N/ATA_HIOW_N 16-bit	(min)	165	125	100	80	70	1
	8-bit	(min)	290	290	290	80	70	
t_{2i}	ATA_ HIOR_N/ATA_HIOW_N recovery time	(min)	-	-	-	70	25	1
t_3	ATA_HIOW_N data setup	(min)	60	45	30	30	20	
t_4	ATA_HIOW_N data hold	min)	30	20	15	10	10	
t_5	ATA_HIOR_N data setup	(min)	50	35	20	20	20	
t_6	ATA_HIOR_N data hold	(min)	5	5	5	5	5	
t_{6Z}	ATA_HIOR_N data tristate	(max)	30	30	30	30	30	2
t_9	ATA_HIOR_N/ATA_HIOW_N to address valid hold	(min)	20	15	10	10	10	
t_{RD}	Read Data Valid to ATA_IORDY active (if ATA_IORDY initially low after t_A)	(min)	0	0	0	0	0	
t_A	ATA_IORDY Setup time		35	35	35	35	35	3
t_B	ATA_IORDY Pulse Width	(max)	1250	1250	1250	1250	1250	
t_C	ATA_IORDY assertion to release	(max)	5	5	5	5	5	

1. t_0 is the minimum total cycle time, t_2 is the minimum command active time, and t_{2i} is the minimum command recovery time or command inactive time. The actual cycle time equals the sum of the actual command active time and the actual command inactive time. The three timing requirements of t_0, t_2, and t_{2i} shall be met. The minimum total cycle time requirements is greater than the sum of t_2 and t_{2i}. This means a host implementation may lengthen either or both t_2 or t_{2i} to ensure that t0 is equal to or greater than the value reported in the devices IDENTIFY DEVICE data. A device implementation shall support any legal host implementation.
2. This parameter specifies the time from the negation edge of ATA_HIOR- to the time that the data bus is no longer driven by the device (tri-state).
3. The delay from the activation of ATA_HIOR- or ATA_HIOW- until the state of ATA_IORDY is first sampled. If ATA_IORDY is inactive then the host shall wait until ATA_IORDY is active before the PIO cycle is completed. If the device is not driving ATA_IORDY negated at the t_A after the activation of ATA_HIOR- or ATA_HIOW-, then t_5 shall be met and t_{RD} is not applicable. If the device is driving ATA_IORDY negated at the time t_A after the activation of ATA_HIOR- or ATA_HIOW-, then t_{RD} shall be met and t_5 is not applicable.



Note 1: ATA_IORDY is not negated for transfer (no wait is generated)

Note 2: ATA_IORDY is negated but is re-assert before t_A (no wait is generated)

Note 3: ATA_IORDY is negated before t_A, and remains asserted until t_B; data is driven valid at t_{RD} (wait is generated)

Fig. 3–22: IDE Interface PIO Data Transfer Timing Diagram

IDE Interface Multiword DMA Timings

Table 3–16: IDE Interface Multiword DMA Timing Parameters

Multiword DMA Timing Parameters (in ns)			Mode 0	Mode 1	Mode 2	Note
t_0	Cycle time	(min)	480	150	120	1
t_D	ATA_HIOR_N/ATA_HIOW_N	(min)	215	80	70	1
t_E	ATA_HIOR_N data access	(max)	150	60	50	
t_F	ATA_HIOR_N data hold	(min)	5	5	5	
t_G	ATA_HIOR_N/ATA_HIOW_N data setup	(min)	100	30	20	
t_H	ATA_HIOW_N data hold	(min)	20	15	10	
t_I	ATA_DACK_N to ATA_HIORN/ATA_HIOWN setup	(min)	0	0	0	
t_J	ATA_HIORN/ATA_HIOWN to ATA_DACK_N hold	(min)	20	5	5	
t_{KR}	ATA_HIORN negated pulse width	(min)	50	50	25	1
t_{KW}	ATA_HIOWN negated pulse width	(min)	215	50	25	1
t_{LR}	ATA_HIORN to ATA_DMARQ delay	(max)	120	40	35	
t_{LW}	ATA_HIOWN to ATA_DMARQ delay	(max)	40	40	35	
t_Z	ATA_DACK_N to tri-state	(max)	20	25	25	

1. t_0 is the minimum total cycle time, t_D is the minimum command active time, and t_K (t_{KR} or t_{KW}, as appropriate) is the minimum command recovery time or command inactive time. The actual cycle time equals the sum of the actual command active time and the actual command inactive time. The three timing requirements of t_0, t_D, t_K shall be met. The minimum total cycle time requirement, t_0, is greater than the sum of t_D and t_K. This means a host implementation may lengthen either or both t_D or t_K to ensure that t_0 is equal to the value reported in the devices IDENTIFY DEVICE data. A device implementation shall support any legal host implementation.

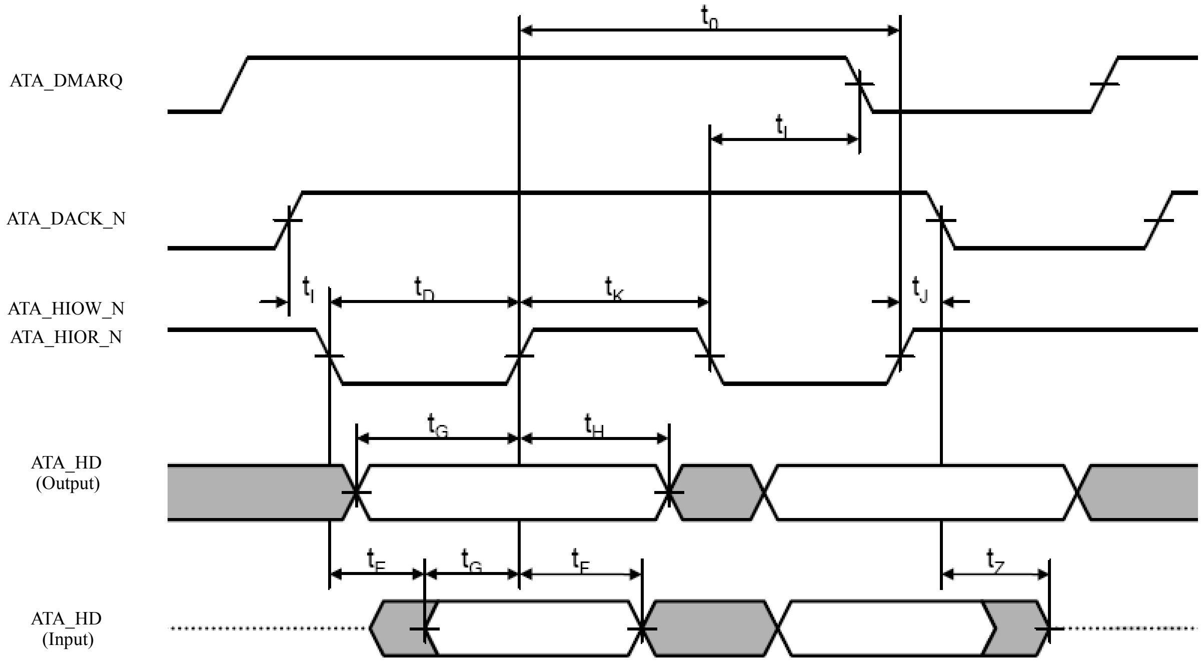


Fig. 3–23: IDE Interface Multiword DMA Timing Diagram.

Ultra DMA Timings

Table 3–17: Ultra DMA Timing Parameters

Ultra DMA Data Burst Timing Requirements

Name	Mode 0 (in ns)		Mode 1 (in ns)		Mode 2 (in ns)		Mode 3 (in ns)		Mode 4 (in ns)		Mode 5* (in ns)		Mode 6* (in ns)		Notes 1
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	
t_{2CYC}	240		160		120		90		60		40		30		S
t_{CYC}	112		73		54		39		25		16.8		13.0		R
t_{2CYC}	230		153		115		86		57		38		29		S
t_{DS}	15.0		10.0		7.0		7.0		5.0		4.0		2.6		R
t_{DH}	5.0		5.0		5.0		5.0		5.0		4.6		3.5		R
t_{DVS}	70.0		48.0		31.0		20.0		6.7		4.8		4.0		S
t_{DVH}	6.2		6.2		6.2		6.2		6.2		4.8		4.0		S
t_{CS}	15.0		10.0		7.0		7.0		5.0		5.0		5.0		D
t_{CH}	5.0		5.0		5.0		5.0		5.0		5.0		5.0		D
t_{CVS}	70.0		48.0		31.0		20.0		6.7		10.0		10.0		H
t_{CVH}	6.2		6.2		6.2		6.2		6.2		10.0		10.0		H
t_{ZFS}	0		0		0		0		0		35		25		D
t_{DZFS}	70.0		48.0		31.0		20.0		6.7		25		17.5		S
t_{FS}		230		200		170		130		120		90		80	D
t_{LI}	0	150	0	150	0	150	0	100	0	100	0	75	0	60	2
t_{MLI}	20		20		20		20		20		20		20		H
t_{UI}	0		0		0		0		0		0		0		H
t_{AZ}		10		10		10		10		10		10		10	3
t_{ZAH}	20		20		20		20		20		20		20		H
t_{ZAD}	0		0		0		0		0		0		0		D
t_{ENV}	20	70	20	70	20	70	20	55	20	55	20	50	20	50	H
t_{RFS}		75		70		60		60		60		50		50	S
t_{RP}	160		125		100		100		100		85		85		R
$t_{IO-RDYZ}$		20		20		20		20		20		20		20	D
$t_{ZI-ORDY}$	0		0		0		0		0		0		0		D

Table 3–17: Ultra DMA Timing Parameters

Ultra DMA Data Burst Timing Requirements															
t_{ACK}	20		20		20		20		20		20		20		H
t_{SS}	50		50		50		50		50		50		50		S

* Ultra DMA Mode 5 and Mode 6 are not supported.

1. All signal transitions for a timing parameter shall be measured at the connector specified in the “Notes” column, where “H” = host, “D” = device, “S” = sender and “R” = recipient. For example, in the case of t_{RFS}, both STROBE and DMARDY- transitions are measured at the sender connector.
2. The parameter t_{LI} shall be measured at the connector of the sender or recipient that is responding to an incoming transition from the recipient or sender respectively. Both the incoming signal and the outgoing response shall be measured at the same connector.
3. The parameter t_{AZ} shall be measured at the connector of the sender or recipient that is releasing the bus.

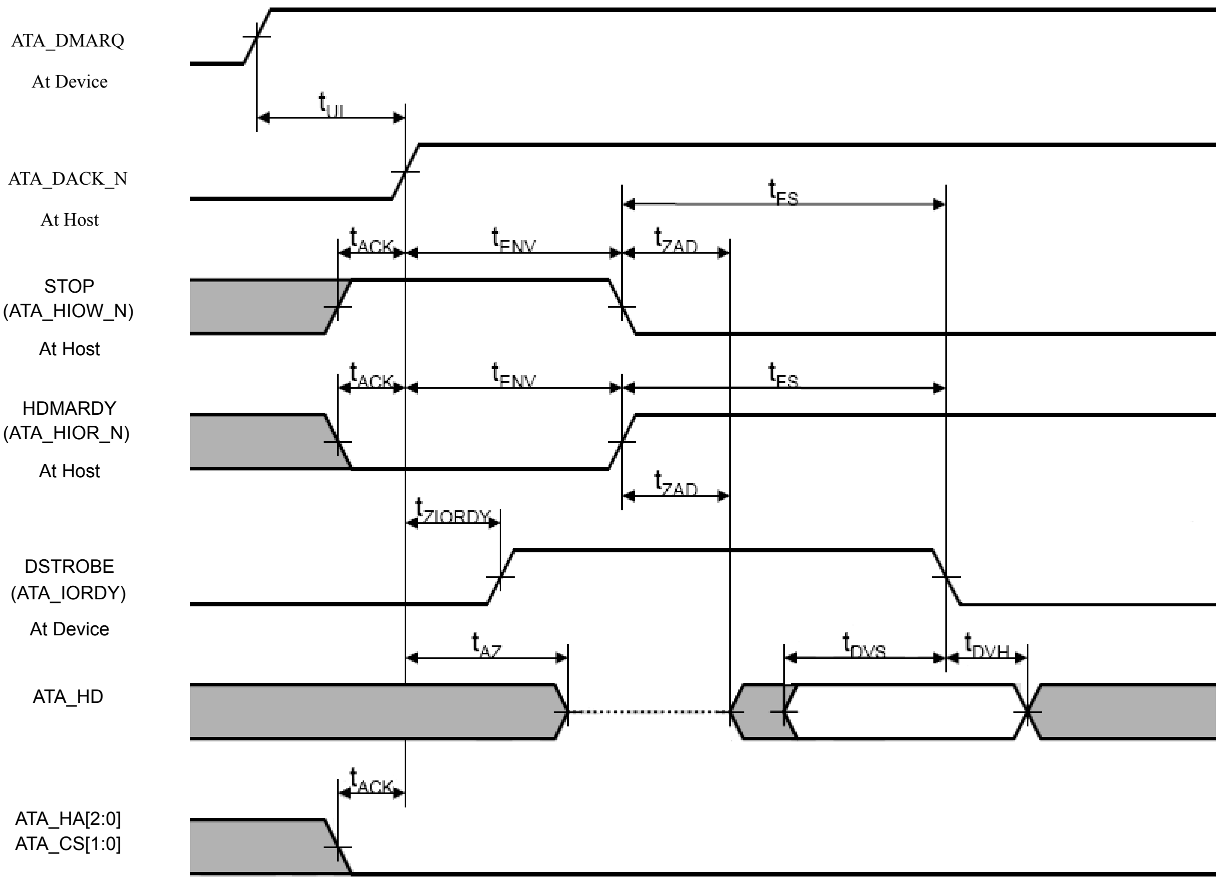
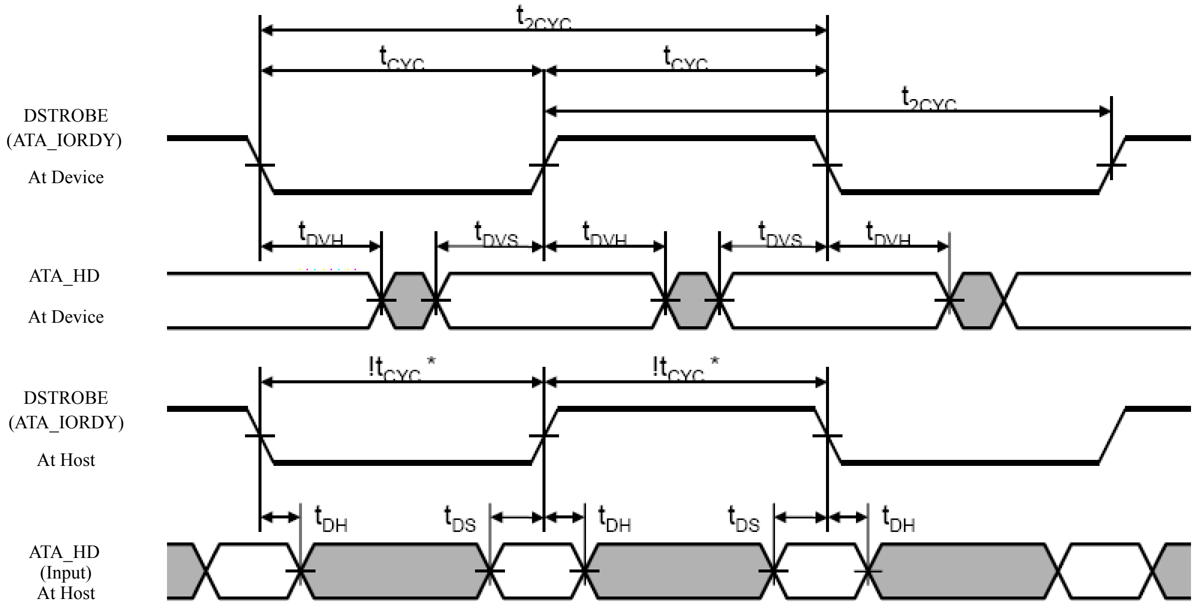


Fig. 3–24: Initiating an Ultra DMA Data-In Burst.



Note: While DSTROBE (**ATA_IORDY**) timing is t_{CYC} at the device, it may be different at the host due to propagation delay differences on the cable.

Fig. 3–25: Sustained Data-In Data Transfer.

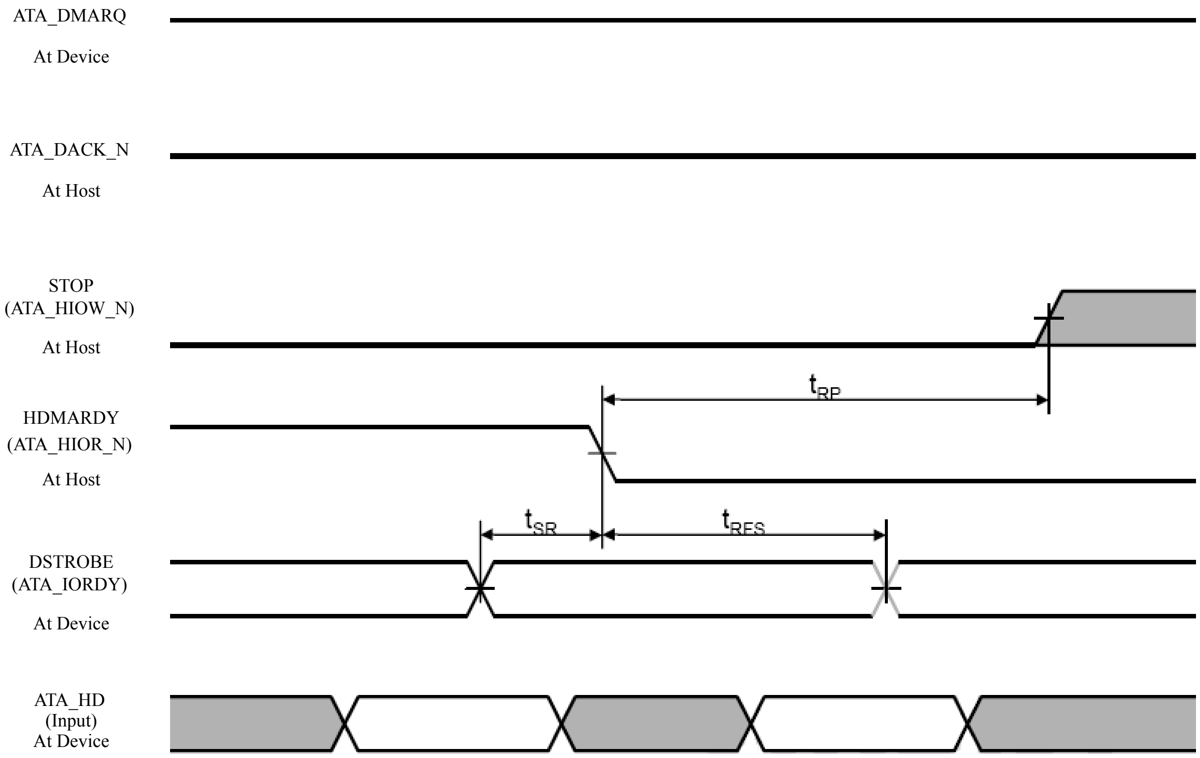


Fig. 3–26: Pausing an Ultra DMA Data- In Burst

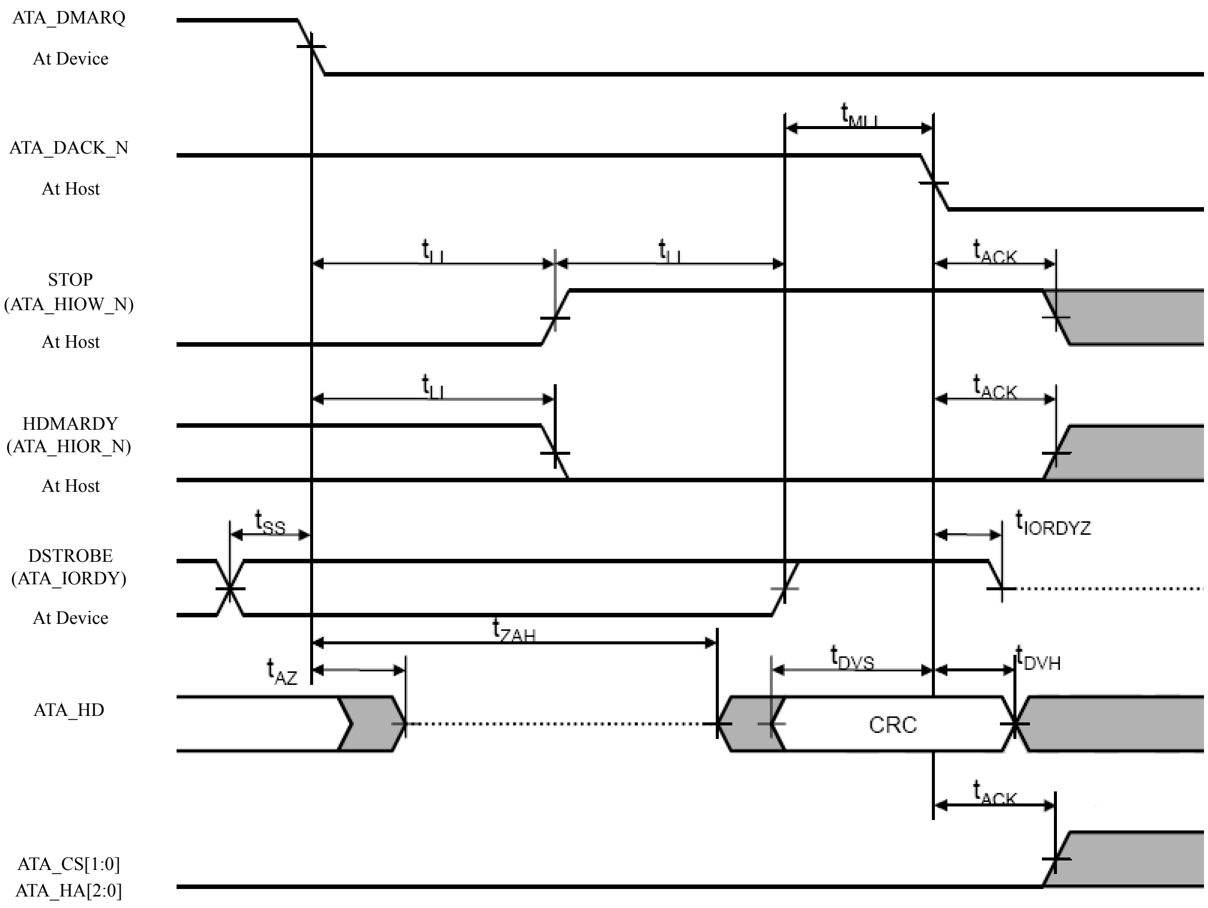


Fig. 3–27: Terminating an Ultra DMA Data-In Burst

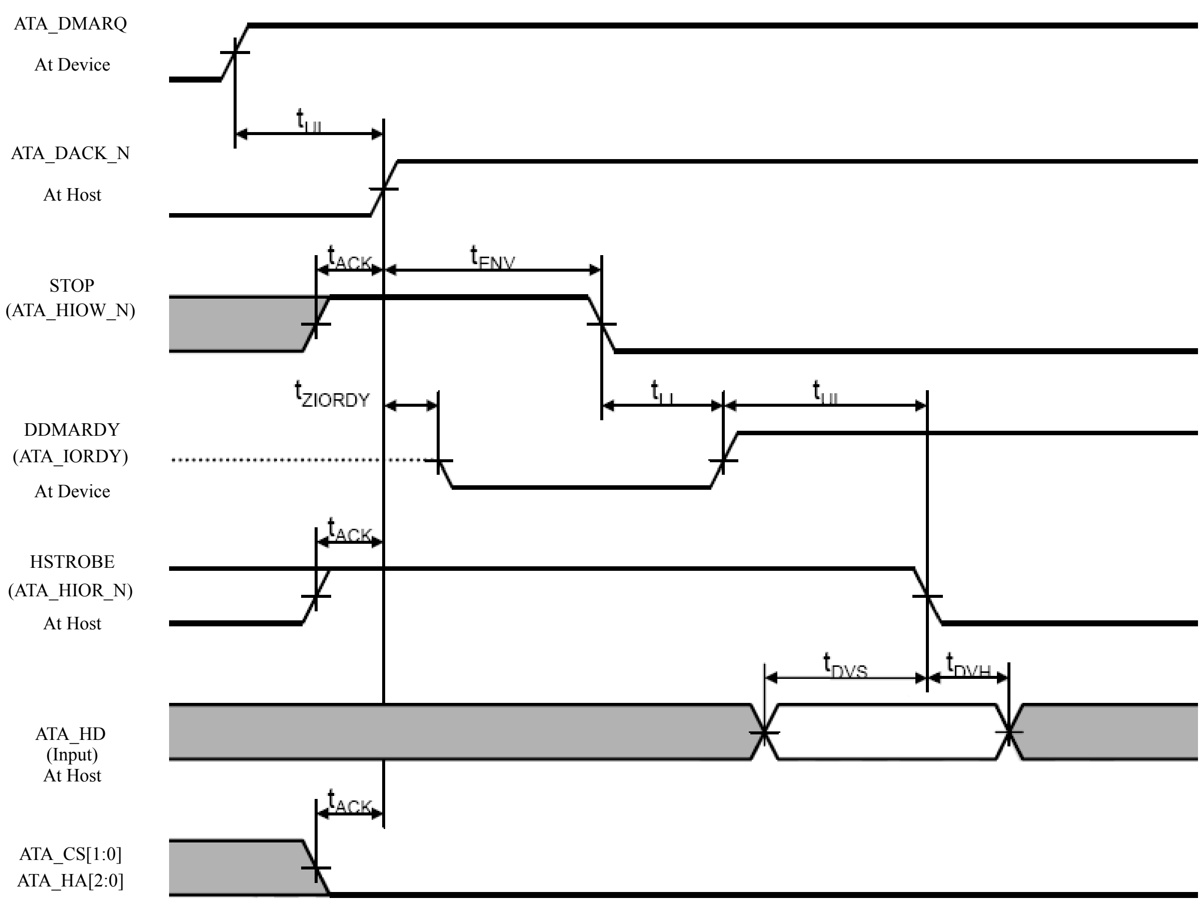
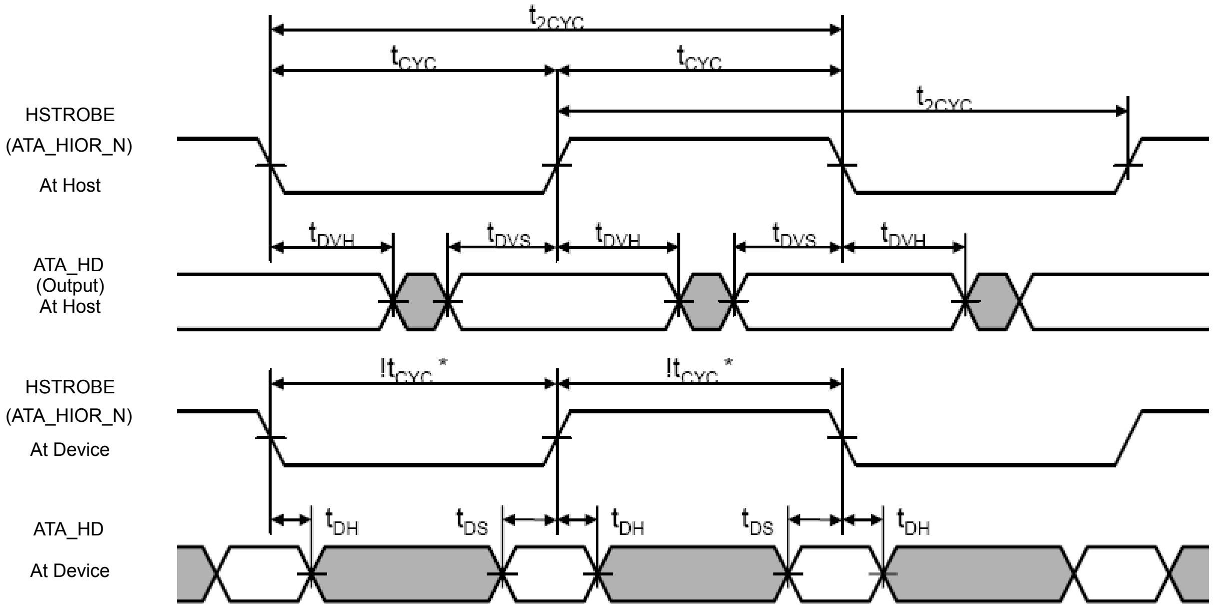


Fig. 3–28: Initiating an Ultra DMA Data-Out Burst.



Note: While HSTROBE timing is t_{CYC} at the host, it may be different at the device due to propagation delay differences on the cable.

Fig. 3–29: Sustained Data-Out Data Transfer.

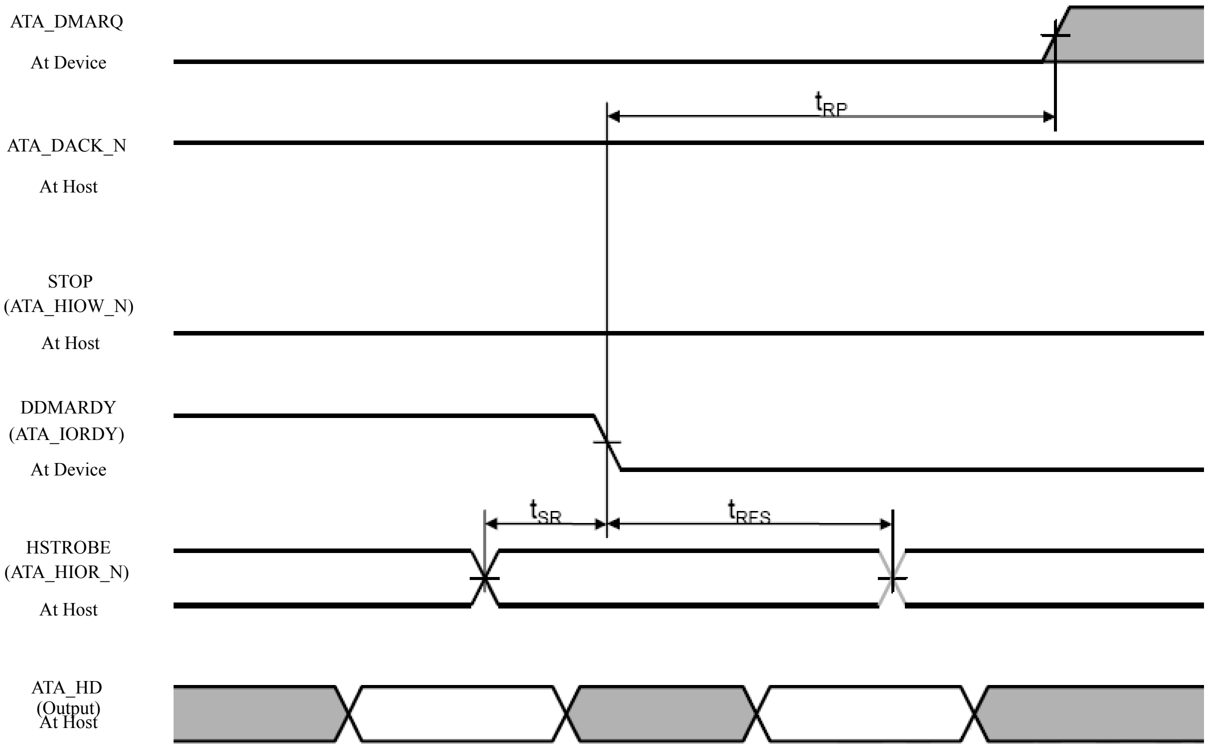


Fig. 3–30: Pausing an Ultra DMA Data-Out Burst

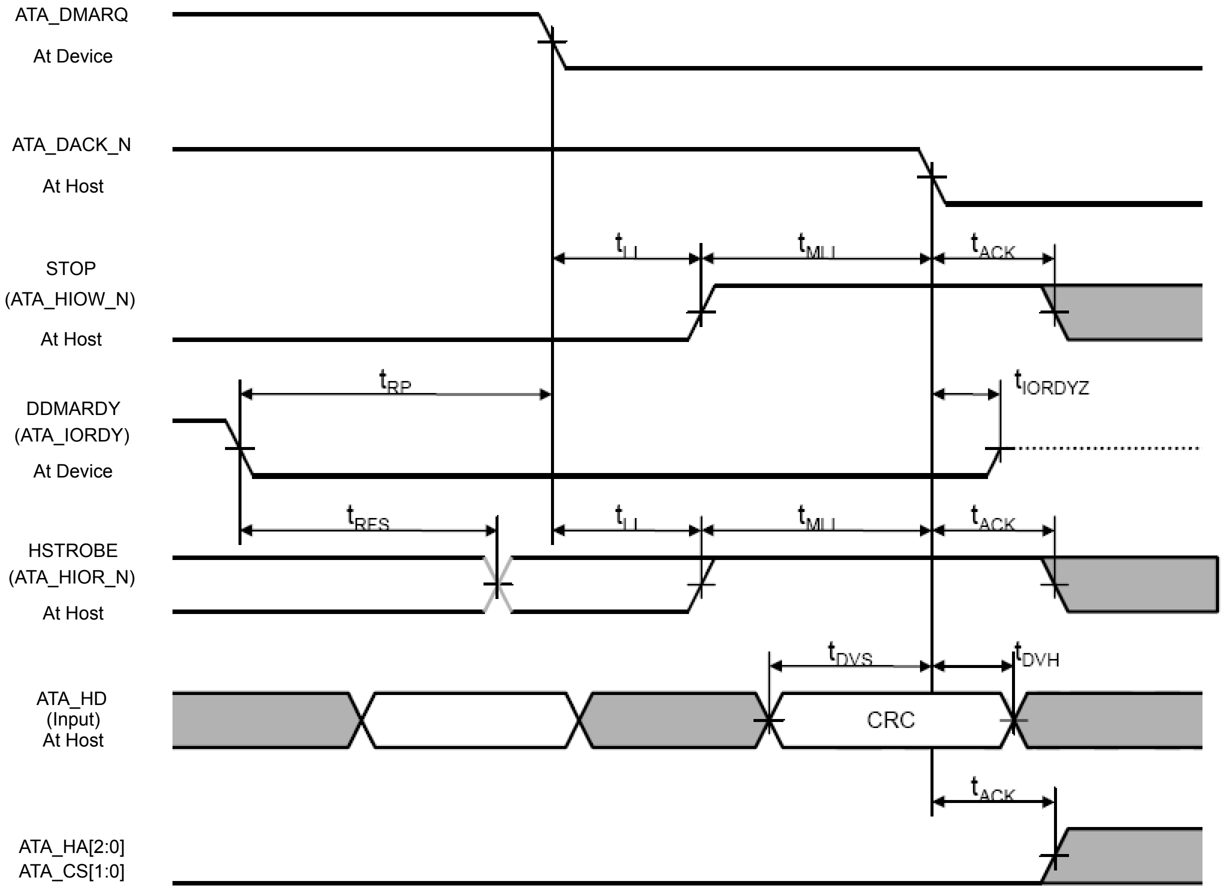


Fig. 3–31: Device Terminating an Ultra DMA Data-Out Burst

3.3. External Pin Interface

This section provides description for each signal in the Cypher ESN7108A. The signals are arranged in functional groups according to the associated interface/ functionality. The N symbol at the end of the signal name indicates the active low state of the signal (asserted when at low voltage level). The absence of N represents the signal as active high (asserted when at the high voltage level). The following notations are used to describe the signal type:

- I = Input Pin
- O = Output Pin
- I/O = Bi-directional Input / Output Pin
- P = Power Pin

3.3.1. Interface Signal Groupings

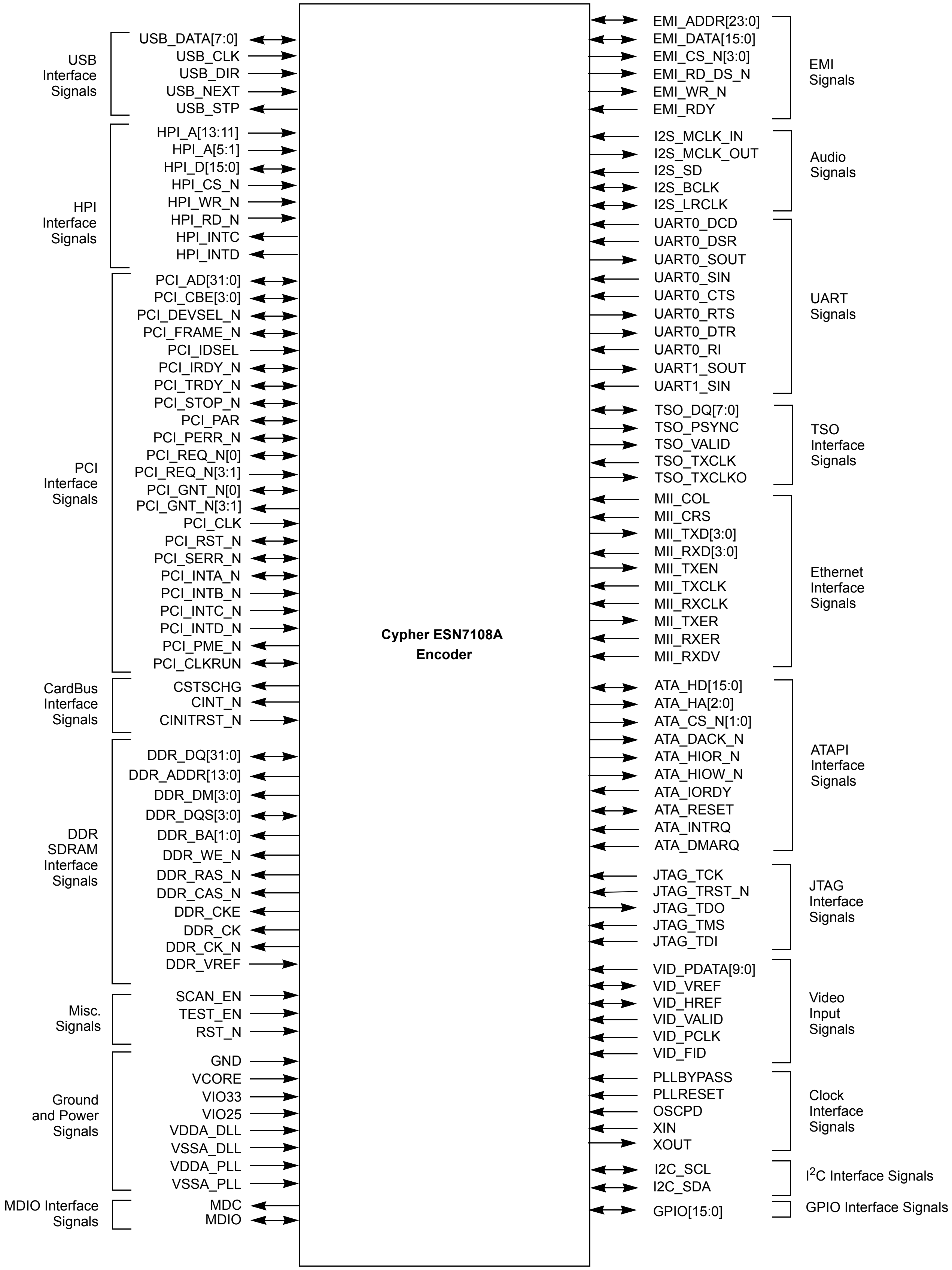


Fig. 3–32: Cypher ESN7108A Interface Signal Groupings

3.3.2. Signal Descriptions

Table 3–18: Cypher ESN7108A Pin Descriptions

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
USB Interface Signals			
USB_DATA[7:0]	I/O		Bidirectional USB data bus.
USB_CLK	I		USB clock input.
USB_DIR	I	PD	USB direction. This signal is driven by the external PHY to determine the direction of USB data flow. 0 = The PHY is ready to receive data from the ESN7108A 1 = The PHY is transferring data to the ESN7108A
USB_NEXT	I	PD	USB next. This signal is asserted by the PHY to throttle all data types except interrupt data and the results of register reads.
USB_STP	O		USB stop. This signal is asserted by the ESN7108A for one clock cycle to signal the end of a USB operation. It can also be used to stop the current receive operation.
HPI Interface Signals			
HPI_A[13:11]	I	PD	HPI address bits 13:11.
HPI_A[5:1]	I	PD	HPI address bits 5:1.
HPI_D[15:0]	I/O	PD	HPI bidirectional data bus.
HPI_CS_N	I	PD	Active low HPI chip select.
HPI_WR_N	I	PD	Active low HPI write strobe.
HPI_RD_N	I	PD	Active low HPI read strobe.
HPI_INTC	O		HPI Control FIFO interrupt. Assertion of this signal interrupts the HPI control FIFO.
HPI_INTD	O		HPI Data FIFO interrupt. Assertion of this signal interrupts the HPI data FIFO.
PCI Interface			
PCI_AD [31:0]	I/O		PCI Address/Data: PCI_AD [31:0] is a multiplexed address and data bus. During the first clock of a transaction, PCI_AD [31:0] contain a physical 32-bit address. During subsequent clocks the PCI_AD [31:0] contains data.
PCI_CBE [3:0]	I/O		Bus Command and Byte Enables: The command and byte enable signals are multiplexed on the same PCI pins. During the address phase of a transaction, PCI_CBE [3:0] define the bus command. During the data phase PCI_CBE [3:0] defines the Byte Enables.

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
PCI_DEVSEL_N	I/O		PCI Device Select: As an output, the ESN7108A asserts PCI_DEVSEL_N when a PCI master peripheral attempts an access to an internal address or an address in the main memory. As an input, PCI_DEVSEL_N indicates the response to a ESN7108A transaction on the PCI bus. PCI_DEVSEL_N is tri-stated signal and remains tri-stated by the ESN7108A until driven by a target device.
PCI_FRAME_N	I/O		Cycle Frame: The current initiator drives PCI_FRAME_N to indicate the beginning and duration of a PCI transaction. While the initiator asserts PCI_FRAME_N, data transfer continues. When the initiator negates PCI_FRAME_N, the transaction is in the final data phase. PCI_FRAME_N is an input to the ESN7108A when it is the target, and PCI_FRAME_N is an output from the ESN7108A when it is the initiator. PCI_FRAME_N remains tri-stated by the ESN7108A until driven by an initiator.
PCI_IDSEL	I		PCI Device Select Configuration transaction. Used as chip select during configuration. Active high.
PCI_IRDY_N	I/O		Initiator Ready: PCI_IRDY_N indicates the ability of ESN7108A as an initiator to complete the current data phase of the transaction. It is used in conjunction with PCI_TRDY_N. When a data phase is completed on any clock, both PCI_IRDY_N and PCI_TRDY_N are sampled asserted. During a write transaction, PCI_IRDY_N indicates the valid data present on PCI_AD [31:0] of the Cypher ESN7108A. During a read transaction, it indicates the readiness of ESN7108A latch data. PCI_IRDY_N is an input to the ESN7108A when it is the target and an output from the ESN7108A when it is an initiator. PCI_IRDY_N remains tri-stated by the ESN7108A until driven by an initiator.
PCI_TRDY_N	I/O		Target Ready: PCI_TRDY_N indicates the ability of ESN7108A as a target to complete the current data phase of the transaction. PCI_TRDY_N is used in conjunction with PCI_IRDY_N. A data phase is completed when both PCI_TRDY_N and PCI_TRDY_N are sampled asserted. During a read transaction PCI_TRDY_N indicates the valid data on the PCI_AD [31:0] placed by ESN7108A as a target. During a write transaction PCI_TRDY_N indicates the readiness of ESN7108A as a target to latch the data. PCI_TRDY_N is an input to the ESN7108A when it is the initiator and an output from the ESN7108A when it is a target. PCI_TRDY_N is tri-stated signal and remains tri-stated by the ESN7108A until driven by a target.

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
PCI_STOP_N	I/O		Stop: PCI_STOP_N indicates that the ESN7108A as a target is requesting the initiator to stop the current transaction. PCI_STOP_N causes the ESN7108A as an initiator to stop the current transaction. PCI_STOP_N is an output when the ESN7108A is a target and an input when the ESN7108A is an initiator.
PCI_PAR	I/O		Calculated/Checked Parity. PAR uses even parity calculated on 36 bits (PCI_AD [31:0] plus PCI_CBE [3:0]. Even parity means that the ESN7108A counts the number of ones within the 36 bits plus PAR and the sum is always even. The ESN7108A always calculates PAR on 36 bits regardless of the valid byte enables. The ESN7108A generates PAR for address and data phases and only guarantees PAR to be valid one PCI clock after the corresponding address or data phase. The ESN7108A drives and tri-states PAR identically to the PCI_AD [31:0] lines except that the ESN7108A delays PAR by exactly one PCI clock. PAR is an output during the address phase (delayed one clock) for all ESN7108A-initiated transactions. PAR is an output during the data phase (delayed one clock) when the ESN7108A is the initiator of a PCI write transaction, and when it is the target of a read transaction. The ESN7108A checks parity when it is the target of a PCI write transaction. If a parity error is detected, the ESN7108A sets the appropriate internal status bits.
PCI_PERR_N	I/O		Parity Error. An external PCI device drives PCI_PERR_N when it receives data that has a parity error. The ESN7108A drives PCI_PERRN when it detects a parity error.
PCI_REQ_N[0]	I/O		PCI Request. In host mode, this pin is an input. In client mode, it is an output.
PCI_REQ_N[3:1]	I	PU	PCI Request. These request pins are only used in host mode.
PCI_GNT_N[0]	I/O		PCI Grant. In host mode, this pin is an output. In client mode, it is an input.
PCI_GNT_N[3:1]	O		PCI Grant. These grant pins are only used in host mode.
PCI_CLK	I		PCI Clock. This is a 33 MHz clock. PCI_CLK provides timing for all transactions on the PCI Bus.
PCI_RST_N	I/O		PCI Reset. This pin is an output in host mode, and an input in client mode.

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
PCI_SERR_N	I/O		System Error. PCI_SERR_N can be pulsed active by any PCI device that detects a system error condition. Upon sampling PCI_SERR_N active, the ESN7108A has the ability to generate an interrupt.
PCI_INTA_N	I/O		PCI interrupt pin. This pin is an input in host mode, and an output in client mode.
PCI_INTB_N PCI_INTC_N PCI_INTD_N	I	PU	PCI interrupt pins. These input-only pins operate only on host mode and contain internal pull-ups.
PCI_PME_N	O		This signal is asserted by the ESN7108A to indicate a change of power management state. The signal is driven based on register information written by the host.
PCI_CLKRUN	I/O		Bidirectional PCI clock run signal.
CardBus Interface Signals			
CSTSCHG	O		Cardbus state change. This signal is driven by the ESN7108A to notify the host about various events. It is an active high asynchronous interrupt that is separate and distinct from the CINT pin.
CINT_N	O		Cardbus interrupt. This signal is generated by the ESN7108A to indicated an interrupt and remains asserted until the interrupt is serviced.
CINITRST_N	I	PU	Cardbus reset.
DDR SDRAM Interface Signals			
DDR_DQ [31:0]	I/O		Data Line. These are the DDR data lines and interface to the SDRAM data bus.
DDR_ADDR [13:0]	O		Memory Address. These signals are used to provide the multiplexed row and column address to the SDRAM
DDR_DM[3:0]	O		Data Mask. When activated during writes, the corresponding data groups in the SDRAM are masked.
DDR_DQS [3:0]	I/O		DDR Data Strobes, one per byte.
DDR_BA [1:0]	O		Bank Address Select. These signals define which banks are selected within each SDRAM rank.
DDR_WE_N	O		Write Enable. DDR_WE_N is used with DDR_CAS_N and DDR_RAS_N along with DDR_CS_N to define the SDRAM commands.

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
DDR_RAS_N	O		Row Address Strobe. This signal is used with DDR_CAS_N and DDR_WE_N along with DDR_CS_N to define the SDRAM commands.
DDR_CAS_N	O		Column Address Strobe. This signal is used with DDR_RAS_N and DDR_WE_N along with DDR_CS_N to define the SDRAM commands.
DDR_CKE	O		Clock Enable. DDR_CKE are used to initialize DDR SDRAM during power-up and to place all SDRAM rows into and out of self-refresh during Suspend-to-RAM. DDR_CKE is also used to dynamically power down inactive SDRAM rows.
DDR_CK	O		Complimentary differential DDR clock. DDR_CK and DDR_CK_N are differential clock output pairs. The crossing of the positive edge of DDR_CK and the negative edge of DDR_CK_N is used to sample the address and control signals on the SDRAM.
DDR_CK_N	O		Complimentary differential DDR clock. DDR_CK and DDR_CK_N are differential clock output pairs. The crossing of the positive edge of DDR_CK and the negative edge of DDR_CK_N is used to sample the address and control signals on the SDRAM.
DDR_VREF	P		1.25V DDR reference voltage.
Extended Memory Interface (EMI)			
EMI_ADDR [23:0]	I/O	PD	EMI Address Bus. Each EMI device has a 21-bit address bit. The additional three lines (EMI_ADDR[23:21]) can be used to decode up to 16 individual Flash devices. Input for Strapping options.
EMI_DATA [7:0]	I/O		EMI Bi-directional Data Bus.
EMI_CS_N[3:0]	O		Active low EMI device chip selects. Within a given address range, up to 4 devices are supported. These pins are connected to the CE# pins of the Flash.
EMI_RD_DS_N	O		Active low EMI device read/data strobe. In Intel® mode, assertion of this pin causes read data to be driven onto the bus by the EMI device. This pin is connected to the OE# pin of the Flash. In Motorola® mode this signal functions as a data strobe.
EMI_WR_N	O		EMI device write. In Intel mode, assertion of this pin causes write data to be driven onto the bus by the ESN7108A. This pin is connected to the WE# pin of the Flash. In Motorola mode, a high voltage on this signal indicates a write operation. A low voltage indicates a read operation.

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
EMI_RDY	I	PU	EMI ready. This pin is driven by the external device to indicate that it is ready to accept the next transaction or data transfer. A low on this pin means (up to 4096 system clock cycles) that the device is not ready to accept data.
Audio Signals			
I2S _MCLK_IN	I		MCLK input.
I2S_MCLK_OUT	O		MCLK output.
I2S_SD	I	PD	Serial Data. This signal represents the serial input audio data.
I2S _BCLK	I/O		BCLK I/O signal. This clock can be configured as input or output in master or slave mode as following: Input - Slave Mode Output - Master Mode
I2S_LRCLK	I/O		Left/ Right Audio Clock. This clock can be configured as input or output in master or slave mode as following: Input - Slave Mode Output - Master Mode
I2C Interface Signals			
I2C_SCL	I/O		I2C clock.
I2C_SDA	I/O		I2C data.
UART Signals			
UART0_DCD	I	PU	UART 0 data carrier detect signal.
UART0_DSR	I	PU	UART 0 data set ready signal.
UART0_SOUT	O		UART 0 serial data out signal.
UART0_SIN	I	PU	UART 0 serial data in signal.
UART0_CTS	I	PU	UART 0 clear to send signal.
UART0_RTS	O		UART 0 request to send signal.
UART0_DTR	O		UART 0 data terminal ready signal.
UART0_RI	I	PU	UART 0 ring indicator signal.
UART1_SOUT	O		UART 1 serial data out. This port does not support flow control.
UART1_SIN	I	PU	UART 1 serial data in. This port does not support flow control.
Transport Stream Output (TSO) Interface Signals			

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
TSO_DQ [7:0]	I/O	PD	TSO data output. When configured in 1-bit output mode, serial data is driven onto TSO_DQ[7]. Input for strapping options.
TSO_PSYNC	O		TSO sync output. This signal is driven by the ESN7108A to indicate the start of a transport packet by signaling the sync byte (0x47).
TSO_VALID	O		Transport stream output data valid. This signal is driven by then ESN7108A to indicate when valid data is on TSO_DQ[7:0].
TSO_TXCLK	I		TSO input clock as one of the TSO output clock sources to be routed to TSO_TXCLKO.
TSO_TXCLKO	O		TSO output clock. Output data on TSO_DQ[7:0] is synchronized to this clock.
MDIO Interface Signals			
MDC	O		Management Data Clock: Clock for the MDIO serial data channel (MDIO). This signal has a maximum frequency of 8 MHz.
MDIO	I/O		Management Data Input/ Output: Bidirectional MDIO data pin for serial communication between the PHY and the ESN7108A.
Ethernet Interface Signals			
MII_COL	I	PD	Collision Detect: This signal is asserted by the PHY when a collision is detected. This signal remains asserted until the collision is resolved. Note that this signal is inactive during full-duplex operation.
MII_CRS	I	PD	Carrier Sense: In half-duplex mode, this signal is asserted by the PHY when transmitting and receiving packets. During full-duplex operation, MII_CRS is only driven by the PHY during a receive operation.
MII_TXD [3:0]	O		Transmit Data: These signals are driven by the ESN7108A when transmitting data to the PHY.
MII_RXD [3:0]	I	PD	Receive Data: These signals are driven by the PHY when transferring data to the ESN7108A.
MII_TXEN	O		Transmit Enable: This signal is asserted by the ESN7108A when valid data is driven onto MII_TXD[3:0].
MII_TXCLK	I		Transmit Clock: The signal is driven at 2.5 MHz during 10 Mbps operation, and at 25 MHz during 100 Mbps operation.
MII_RXCLK	I		Receive Clock: The signal is driven at 2.5 MHz during 10 Mbps operation, and at 25 MHz during 100 Mbps operation.

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
MII_TXER	O		Transmit Error: This signal is asserted by the ESN7108A to indicate an error during the transmit operation. This signal must be synchronized to MII_TXCLK.
MII_RXER	I	PD	Receive Error: This signal is asserted by the PHY to indicate an error during the receive operation. This signal must be synchronized to MII_RXCLK.
MII_RXDV	I	PD	Receive Date Valid: This signal is asserted by the PHY to indicate when valid data is driven onto MII_RXD[3:0].
ATAPI Interface Signals			
ATA_HD[15:0]	I/O	PD	ATAPI bidirectional data pins.
ATA_HA[2:0]	O		ATAPI interface address pins.
ATA_CS_N[1:0]	O		ATAPI interface active low chip selects.
ATA_DACK_N	O		Active low DMA acknowledge signal.
ATA_HIOR_N	O		Active low ATAPI interface read signal.
ATA_HIOW_N	O		Active low ATAPI interface write signal.
ATA_IORDY	I	PD	Ready signal. This pin is driven by the hard disk to indicate it is ready to accept data.
ATA_RESET	I/O	PD	Bidirectional ATAPI interface reset signal.
ATA_INTRQ	I	PD	ATAPI interface interrupt signal.
ATA_DMARQ	I	PD	DMA request signal. This signal is driven by external logic to request a DMA transfer.
GPIO Interface Signals			
GPIO[7:0]	I/O		Programmable GPIO signal. Each GPIO pin can be used for general purpose communication and can be configured as either an input or an output by software via the register interface.
JTAG Interface Signals			
JTAG_TCK	I	PU	JTAG Test Clock. Can be left open if not used.
JTAG_TRST_N	I	PD	JTAG Test Reset. Can be left open if not used.
JTAG_TDO	O		JTAG Test Data Output.
JTAG_TMS	I	PU	JTAG Test Mode Select. Can be left open if not used.
JTAG_TDI	I	PU	JTAG Test Data Input. Can be left open if not used.

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
Video Input Signals			
VID_PDATA[9:0]	I		Pixel Data Input
VID_VREF	I/O	PD	Vertical Reference. Active edge on this pin indicates beginning of active video data in a vertical period. The active edge can be configured as a positive or negative edge.
VID_HREF	I/O	PD	Horizontal Reference. Active edge on this pin indicates beginning of active video data in a horizontal period. The active edge can be configured as a positive or negative edge.
VID_VALID	I	PU	Valid. This signal represents the valid video pixel data.
VID_PCLK	I		Video Pixel Clock:
VID_FID	I	PD	Field ID for interlaced video.
Clock Interface Signals			
PLLBYPASS	I	PD	Assertion of this pin by external logic bypasses the PLL.
PLLRESET	I	PD	PLL reset. Assertion of this pin by external logic resets the PLL.
OSCPD	I	PD	Crystal oscillator power down. When asserted, the internal oscillator circuit is powered down.
XIN	I		33 MHz crystal oscillator input.
XOUT	O		33 MHz crystal oscillator output.
Miscellaneous Signals			
SCAN_EN	I	PD	Scan enable.
TEST_EN	I	PD	Test enable.
RST_N	I	PU	Active low master reset. Assertion of this signal resets the entire device.
Ground and Power Signals			
GND	P		Ground for the ESN7108A core logic and I/O.
VCORE	P		1.25V power supply for the ESN7108A core logic.
VIO33	P		3.3V power supply for the ESN7108A 3.3V I/O.
VIO25	P		2.5V power supply for the ESN7108A 2.5V I/O.
VDDA_DLL	P		Analog 1.25V power pins for the internal clock generation circuit.
VSSA_DLL	P		Analog ground pins for the internal clock generation circuit.

Table 3–18: Cypher ESN7108A Pin Descriptions, continued

Name	Type	Internal Pull Up (PU) /Pull Down (PD)	Description
VDDA_PLL	P		Analog 1.25V PLL power pin.
VSSA_PLL	P		Analog PLL ground pin.

3.3.3. Strapping Options

Table 3–19 shows the strapping options for the Cypher ESN7108A. These options are selected by setting certain pins to a known state during reset as shown in the table below.

Table 3–19: ESN7108A Strapping Options

Pin Name	Default	Definition
EMI Address Pins		
EMI_ADDR[0]	0	0 - CPUs will boot from flash ROM 1 - CPUs will boot from Power On Self Test (POST) ROM
EMI_ADDR[1]	0	Reserved
EMI_ADDR[2]	0	Reserved
EMI_ADDR[3]	0	0 - The ESN7108A operates in PCI client mode 1 - The ESN7108A operates in PCI Host mode.
EMI_ADDR[4]	0	0 - IDE interface is disabled. 1 - IDE interface is enabled.
EMI_ADDR[5]	0	0 - IDE clock operates at 33 MHz 1 - IDE clock operates at 66 MHz
EMI_ADDR[6]	0	0 - IDE interface conforms to standard IDE interface timing 1 - IDE interface conforms to Intel timing
EMI_ADDR[7]	0	0 - HPI interface is disabled. 1 - HPI interface is enabled.
EMI_ADDR[8]	0	0 - HPI interface conforms to Motorola timing. 1 - HPI interface conforms to Intel timing.
EMI_ADDR[9]	0	0 - USB interface is disabled. 1 - USB interface is enabled.
EMI_ADDR[11:10]	0	Selects the encoder clock frequency (based on 33 MHz f_{XIN}) 00 - 133 MHz 01 - 111 MHz 10 - 83 MHz 11 - 55.5 MHz
EMI_ADDR[15:12]	0	Reserved

Table 3–19: ESN7108A Strapping Options, continued

Pin Name	Default	Definition
EMI_ADDR[23:16]	0	Reserved for Firmware use. See Section 3.3.3.1. "FW_BSP Strap Options" on page 81.
Transport Stream Output (TSO) Data Pins		
TSO_DQ[0]	0	IDE primary cable select: 0 - 40-pin cable 1 - 80-pin cable
TSO_DQ[7:1]	0	Reserved

3.3.3.1. FW_BSP Strap Options

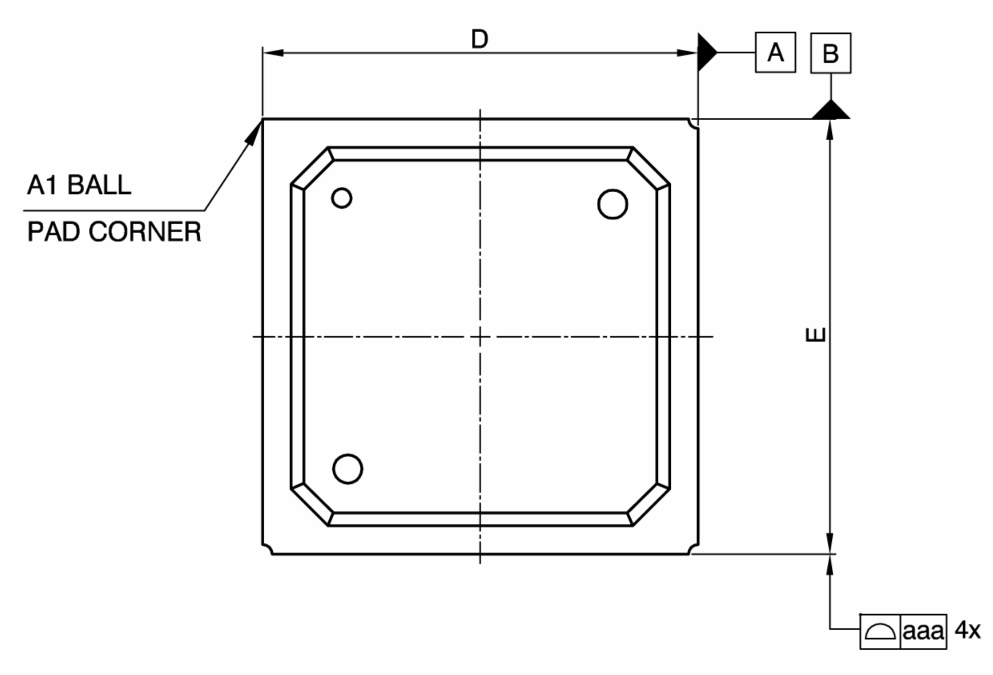
The following options in the CCB.FW_BSP field are controlled by hardware strap options and used by firmware to configure system software:

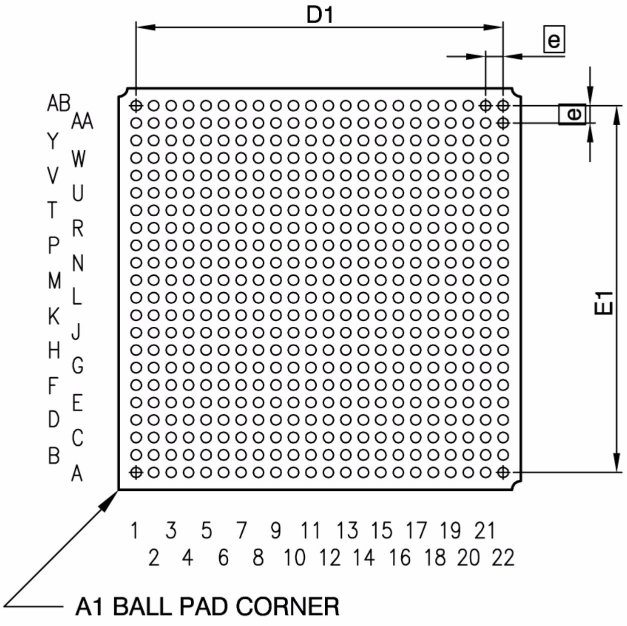
Table 3–20:

Pin Name	Definition
FW_BSP[7:4]	SOC Core Clock 0000 - 10Mhz 0000 - 10Mhz 0001 - 16Mhz 0010 - 25Mhz 0011 - 75Mhz 0100 - 100Mhz 0101 - 111Mhz 0110 - 115Mhz 0111 - 125Mhz 1000 - 133Mhz 1001 - 150Mhz 1010 - 166Mhz 1011 - 170Mhz 1100 - 175Mhz 1101 - 180Mhz 1110 - 185Mhz 1111 - 200Mhz
FW_BSP[2:0]	POST Rom Options 000 - Host 001 - PCI Boot Mode (Also implies PCI Client strapped) 010 - CLI or XModem Boot Mode (COM0=CPU0, COM1 =CPU1), 57600baud 011 - CLI or XModem Boot Mode (COM0=CPU0, COM1=CPU1), 9600baud 100 - Run POST on CPU1 (MFG Test mode) 101 - POST PCI Boot 110 - POST CLI or 576000 XModem Boot 111 - POST CLI or 9600 XModem Boot
FW_BSP[3]	DDR x16/x32 1xxx - DDRx32 mode 0xxx - DDRx16 mode

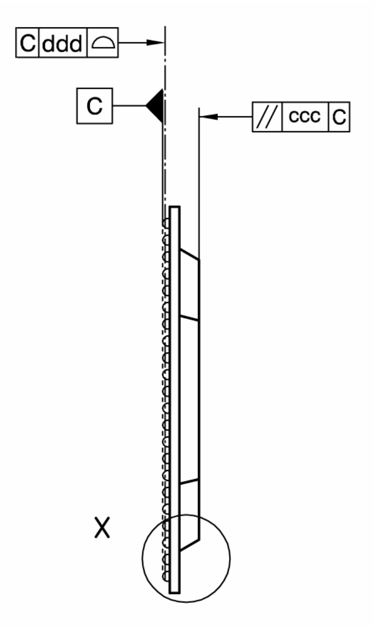
3.4. Mechanical Specifications

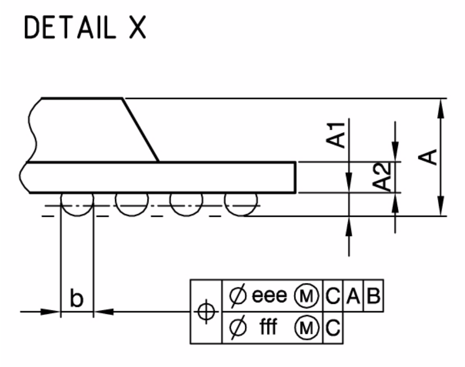
UCBGA1 Package Specifications





UNIT	A	A1	A2	b	D	D1	E	E1	e	aaa	ccc	ddd	eee	fff
mm	2.44 2.02	MIN 0.3	0.62 0.50	0.7 0.5	23.0	21.0	23.0	21.0	1.0	0.20	0.35	0.20	0.25	0.10





3.5. Thermal Specifications

Table 3–21: Thermal Specifications

Parameter	Symbol	Min	Typ	Max	Unit	Note
Thermal resistance from junction to ambient.	θ_{ja}		22.2		°C/W	$\theta_{ja}=(T_j-T_a)/P$
Thermal resistance from junction to case.	θ_{jc}		8.6		°C/W	$\theta_{jc}=(T_j-T_c)/P$
Thermal characterization parameter from junction to package top center.	Ψ_{jt}		1.4		°C/W	May be used to **estimate** die temperature by measuring the top center of the package. $\Psi_{jt}=(T_j-T_t)/P$

Where,
T_j = Junction Temperature
T_t = Package Top center temperature
T_c = Case Temperature
P = Power dissipation

3.6. Numerical Pin Listing

Table 3–22: Cypher ESN7108A Numerical Pin Listing

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
A1	VIO33	B9	DDR_ADDR[2]	C17	DDR_DQ[6]
A2	GND	B10	GND	C18	DDR_DQ[2]
A3	VIO25	B11	DDR_ADDR[8]	C19	GND
A4	DDR_DQ[31]	B12	DDR_RAS_N	C20	EMI_CS_N[1]
A5	DDR_DQ[28]	B13	GND	C21	EMI_CS_N[2]
A6	DDR_DQ[24]	B14	DDR_ADDR[13]	C22	EMI_CS_N[3]
A7	VIO25	B15	DDR_DQ[13]	D1	VID_PDATA[9]
A8	DDR_DQS[2]	B16	GND	D2	VID_FID
A9	DDR_ADDR[1]	B17	DDR_DM[0]	D3	VID_HREF
A10	VIO25	B18	DDR_DQ[3]	D4	GND
A11	DDR_CK	B19	GND	D5	DDR_DQ[26]
A12	DDR_CK_N	B20	EMI_WR_N	D6	DDR_DQ[22]
A13	VIO25	B21	EMI_RD_DS_N	D7	DDR_DQ[19]
A14	DDR_ADDR[12]	B22	EMI_RDY	D8	DDR_DM[2]
A15	DDR_DQ[14]	C1	VID_PCLK	D9	DDR_ADDR[4]
A16	VIO25	C2	VID_VREF	D10	DDR_ADDR[7]
A17	DDR_DM[1]	C3	GND	D11	DDR_CAS_N
A18	DDR_DQS[0]	C4	DDR_DQ[29]	D12	DDR_ADDR[9]
A19	DDR_DQ[7]	C5	DDR_DQ[27]	D13	DDR_ADDR[11]
A20	VIO25	C6	DDR_DQ[23]	D14	DDR_BA[1]
A21	GND	C7	DDR_DQ[20]	D15	DDR_DQS[1]
A22	VIO33	C8	DDR_DQ[16]	D16	DDR_DQ[9]
B1	VID_VALID	C9	DDR_ADDR[3]	D17	DDR_DQ[5]
B2	OSCPD	C10	DDR_ADDR[6]	D18	DDR_DQ[1]
B3	GND	C11	DDR_CKE	D19	GND
B4	DDR_DQ[30]	C12	DDR_WE_N	D20	EMI_DQ[6]
B5	DDR_DQS[3]	C13	DDR_ADDR[10]	D21	EMI_DQ[7]
B6	DDR_DM[3]	C14	DDR_BA[0]	D22	EMI_CS_N[0]
B7	GND	C15	DDR_DQ[12]	E1	VID_PDATA[6]
B8	DDR_DQ[17]	C16	DDR_DQ[10]	E2	VID_PDATA[7]

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
E3	VID_PDATA[8]	F11	DDR_VREF	G19	EMI_ADDR[19]
E4	GND	F12	GND	G20	EMI_ADDR[20]
E5	DDR_DQ[25]	F13	VSSA_DLL	G21	EMI_ADDR[21]
E6	DDR_DQ[21]	F14	VDDA_DLL	G22	EMI_ADDR[22]
E7	DDR_DQ[18]	F15	GND	H1	XIN
E8	DDR_ADDR[0]	F16	VSSA_DLL	H2	GPIO[3]
E9	DDR_ADDR[5]	F17	GND	H3	GPIO[4]
E10	VIO25	F18	GND	H4	GPIO[5]
E11	VIO25	F19	EMI_ADDR[23]	H5	GPIO[6]
E12	VIO25	F20	EMI_DQ[0]	H6	GPIO[7]
E13	VIO25	F21	EMI_DQ[1]	H7	VIO33
E14	DDR_DQ[15]	F22	EMI_DQ[2]	H8	GND
E15	DDR_DQ[11]	G1	VIO33	H9	VCORE
E16	DDR_DQ[8]	G2	GND	H10	GND
E17	DDR_DQ[4]	G3	VID_PDATA[0]	H11	GND
E18	DDR_DQ[0]	G4	VID_PDATA[1]	H12	GND
E19	GND	G5	VID_PDATA[2]	H13	GND
E20	EMI_DQ[3]	G6	VID_PDATA[3]	H14	VCORE
E21	EMI_DQ[4]	G7	VDDA_DLL	H15	GND
E22	EMI_DQ[5]	G8	GND	H16	VIO33
F1	VID_PDATA[4]	G9	VCORE	H17	EMI_ADDR[13]
F2	VID_PDATA[5]	G10	VCORE	H18	EMI_ADDR[14]
F3	PLLRESET	G11	VCORE	H19	EMI_ADDR[15]
F4	I2C_SCL	G12	VCORE	H20	EMI_ADDR[16]
F5	I2C_SDA	G13	VCORE	H21	GND
F6	GND	G14	VCORE	H22	VIO33
F7	VSSA_DLL	G15	GND	J1	XOUT
F8	GND	G16	VDDA_DLL	J2	I2SI_SD
F9	VDDA_DLL	G17	EMI_ADDR[17]	J3	TEST_EN
F10	VSSA_DLL	G18	EMI_ADDR[18]	J4	GPIO[0]

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
J5	GPIO[1]	K13	GND	L21	TSO_TXCLK
J6	GPIO[2]	K14	VCORE	L22	EMI_ADDR[0]
J7	VIO33	K15	GND	M1	MII_TXER
J8	GND	K16	VIO33	M2	MII_TXCLK
J9	VCORE	K17	EMI_ADDR[1]	M3	MII_COL
J10	GND	K18	EMI_ADDR[2]	M4	MII_CRS
J11	GND	K19	EMI_ADDR[3]	M5	MII_RXCLK
J12	GND	K20	EMI_ADDR[4]	M6	MII_RXD[0]
J13	GND	K21	EMI_ADDR[5]	M7	VDDA_PLL
J14	VCORE	K22	EMI_ADDR[6]	M8	GND
J15	GND	L1	MII_RXD[1]	M9	VCORE
J16	VIO33	L2	MII_RXD[2]	M10	GND
J17	EMI_ADDR[7]	L3	MII_RXD[3]	M11	GND
J18	EMI_ADDR[8]	L4	GND	M12	GND
J19	EMI_ADDR[9]	L5	MII_RXER	M13	GND
J20	EMI_ADDR[10]	L6	MII_RXDV	M14	VCORE
J21	EMI_ADDR[11]	L7	VSSA_PLL	M15	GND
J22	EMI_ADDR[12]	L8	GND	M16	VIO33
K1	SCAN_EN	L9	VCORE	M17	TSO_DQ[1]
K2	I2S_BCLK	L10	GND	M18	TSO_DQ[2]
K3	I2S_LRCLK	L11	GND	M19	GND
K4	I2S_MCLK_IN	L12	GND	M20	TSO_DQ[3]
K5	PLLBYPASS	L13	GND	M21	TSO_DQ[4]
K6	I2S_MCLK_OUT	L14	VCORE	M22	TSO_DQ[5]
K7	VIO33	L15	GND	N1	MDIO
K8	GND	L16	VIO33	N2	MII_TXD[0]
K9	VCORE	L17	TSO_DQ[6]	N3	MII_TXD[1]
K10	GND	L18	TSO_DQ[7]	N4	MII_TXD[2]
K11	GND	L19	TSO_TXCLKO	N5	MII_TXD[3]
K12	GND	L20	TSO_VALID	N6	MII_TXEN

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
N7	VIO33	P15	GND	T1	ATA_HD[1]
N8	GND	P16	VIO33	T2	ATA_HD[0]
N9	VCORE	P17	HPI_D[14]	T3	ATA_HA[2]
N10	GND	P18	HPI_D[15]	T4	ATA_HA[1]
N11	GND	P19	HPI_A[1]	T5	ATA_HA[0]
N12	GND	P20	HPI_A[2]	T6	ATA_CS_N[1]
N13	GND	P21	HPI_A[3]	T7	VIO33
N14	VCORE	P22	HPI_A[4]	T8	GND
N15	GND	R1	VIO33	T9	GND
N16	VIO33	R2	GND	T10	GND
N17	HPI_A[5]	R3	ATA_CS_N[0]	T11	GND
N18	HPI_A[11]	R4	ATA_DACK_N	T12	GND
N19	HPI_A[12]	R5	ATA_HIOR_N	T13	GND
N20	HPI_A[13]	R6	ATA_HIOW_N	T14	GND
N21	TSO_PSYNC	R7	VIO33	T15	GND
N22	TSO_DQ[0]	R8	GND	T16	VIO33
P1	UART0_DSR	R9	VCORE	T17	HPI_D[4]
P2	UART0_CTS	R10	VCORE	T18	HPI_D[5]
P3	UART0_DCD	R11	VCORE	T19	HPI_D[6]
P4	UART0_DTR	R12	VCORE	T20	HPI_D[7]
P5	RST_N	R13	VCORE	T21	GND
P6	MDC	R14	VCORE	T22	VIO33
P7	VIO33	R15	GND	U1	ATA_HD[7]
P8	GND	R16	VIO33	U2	ATA_HD[6]
P9	VCORE	R17	HPI_D[8]	U3	ATA_HD[5]
P10	GND	R18	HPI_D[9]	U4	ATA_HD[4]
P11	GND	R19	HPI_D[10]	U5	ATA_HD[3]
P12	GND	R20	HPI_D[11]	U6	ATA_HD[2]
P13	GND	R21	HPI_D[12]	U7	VIO33
P14	VCORE	R22	HPI_D[13]	U8	VIO33

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
U9	VIO33	V17	PCI_GNT_N[3]	Y3	ATA_INTRQ
U10	VIO33	V18	PCI_AD[2]	Y4	UART0_SIN
U11	VIO33	V19	CINT_N	Y5	EJ_TDI
U12	VIO33	V20	HPI_CS_N	Y6	PCI_INTC_N
U13	VIO33	V21	HPI_INTC	Y7	PCI_AD[31]
U14	VIO33	V22	HPI_INTD	Y8	PCI_CBE[3]
U15	VIO33	W1	ATA_HD[11]	Y9	PCI_AD[25]
U16	VIO33	W2	ATA_HD[13]	Y10	PCI_CBE[2]
U17	HPI_RD_N	W3	ATA_IORDY	Y11	PCI_AD[18]
U18	HPI_WR_N	W4	UART0_RTS	Y12	PCI_IRDY_N
U19	HPI_D[0]	W5	EJ_TRST_N	Y13	PCI_STOP_N
U20	HPI_D[1]	W6	PCI_INTA_N	Y14	PCI_AD[8]
U21	HPI_D[2]	W7	PCI_REQ_N[0]	Y15	PCI_AD[6]
U22	HPI_D[3]	W8	PCI_IDSEL	Y16	PCI_GNT_N[2]
V1	ATA_HD[10]	W9	PCI_AD[26]	Y17	PCI_CBE[0]
V2	ATA_HD[9]	W10	PCI_AD[21]	Y18	PCI_AD[0]
V3	ATA_HD[8]	W11	PCI_AD[19]	Y19	CSTSCHG
V4	UART1_SOUT	W12	PCI_AD[15]	Y20	USB_STP
V5	EJ_TDO	W13	PCI_AD[11]	Y21	USB_DATA[2]
V6	PCI_INTB_N	W14	PCI_AD[9]	Y22	USB_DATA[4]
V7	PCI_GNT_N[0]	W15	PCI_SERR_N	AA1	GND
V8	PCI_AD[28]	W16	PCI_REQ_N[1]	AA2	ATA_HD[15]
V9	PCI_AD[27]	W17	PCI_AD[5]	AA3	ATA_DMARQ
V10	PCI_AD[22]	W18	PCI_AD[1]	AA4	UART1_SIN
V11	PCI_AD[20]	W19	CINITRST_N	AA5	EJ_TCK
V12	PCI_CLK	W20	USB_CLK	AA6	PCI_INTD_N
V13	PCI_AD[12]	W21	USB_DATA[6]	AA7	PCI_AD[30]
V14	PCI_AD[10]	W22	USB_DATA[7]	AA8	GND
V15	PCI_PERR_N	Y1	ATA_HD[12]	AA9	PCI_AD[24]
V16	PCI_GNT_N[1]	Y2	ATA_HD[14]	AA10	PCI_TRDY_N

Pin No.	Pin Name	Pin No.	Pin Name
AA11	PCI_AD[17]	AB19	USB_NEXT
AA12	PCI_AD[14]	AB20	USB_DATA[1]
AA13	PCI_DEVSEL_N	AB21	GND
AA14	PCI_AD[7]	AB22	VIO33
AA15	GND		
AA16	PCI_REQ_N[2]		
AA17	PCI_AD[4]		
AA18	PCI_PME_N		
AA19	USB_DIR		
AA20	USB_DATA[0]		
AA21	USB_DATA[3]		
AA22	USB_DATA[5]		
AB1	VIO33		
AB2	ATA_RESET		
AB3	UART0_SOUT		
AB4	UART0_RI		
AB5	EJ_TMS		
AB6	PCI_RST_N		
AB7	PCI_AD[29]		
AB8	VIO33		
AB9	PCI_AD[23]		
AB10	PCI_FRAME_N		
AB11	PCI_AD[16]		
AB12	PCI_AD[13]		
AB13	PCI_CBE[1]		
AB14	PCI_PAR		
AB15	VIO33		
AB16	PCI_REQ_N[3]		
AB17	PCI_AD[3]		
AB18	PCI_CLKRUN		

3.7. Alphabetical Pin Listings

Table 3–23: Cypher ESN7108A Alphabetical Pinout

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
R3	ATA_CS_N[0]	E8	DDR_ADDR[0]	A19	DDR_DQ[7]
T6	ATA_CS_N[1]	A9	DDR_ADDR[1]	E16	DDR_DQ[8]
R4	ATA_DACK_N	B9	DDR_ADDR[2]	D16	DDR_DQ[9]
AA3	ATA_DMARQ	C9	DDR_ADDR[3]	C16	DDR_DQ[10]
T5	ATA_HA[0]	D9	DDR_ADDR[4]	E15	DDR_DQ[11]
T4	ATA_HA[1]	E9	DDR_ADDR[5]	C15	DDR_DQ[12]
T3	ATA_HA[2]	C10	DDR_ADDR[6]	B15	DDR_DQ[13]
T2	ATA_HD[0]	D10	DDR_ADDR[7]	A15	DDR_DQ[14]
T1	ATA_HD[1]	B11	DDR_ADDR[8]	E14	DDR_DQ[15]
U6	ATA_HD[2]	D12	DDR_ADDR[9]	C8	DDR_DQ[16]
U5	ATA_HD[3]	C13	DDR_ADDR[10]	B8	DDR_DQ[17]
U4	ATA_HD[4]	D13	DDR_ADDR[11]	E7	DDR_DQ[18]
U3	ATA_HD[5]	A14	DDR_ADDR[12]	D7	DDR_DQ[19]
U2	ATA_HD[6]	B14	DDR_ADDR[13]	C7	DDR_DQ[20]
U1	ATA_HD[7]	C14	DDR_BA[0]	E6	DDR_DQ[21]
V3	ATA_HD[8]	D14	DDR_BA[1]	D6	DDR_DQ[22]
V2	ATA_HD[9]	D11	DDR_CAS_N	C6	DDR_DQ[23]
V1	ATA_HD[10]	A11	DDR_CK	A6	DDR_DQ[24]
W1	ATA_HD[11]	C11	DDR_CKE	E5	DDR_DQ[25]
Y1	ATA_HD[12]	A12	DDR_CK_N	D5	DDR_DQ[26]
W2	ATA_HD[13]	B17	DDR_DM[0]	C5	DDR_DQ[27]
Y2	ATA_HD[14]	A17	DDR_DM[1]	A5	DDR_DQ[28]
AA2	ATA_HD[15]	D8	DDR_DM[2]	C4	DDR_DQ[29]
R5	ATA_HIOR_N	B6	DDR_DM[3]	B4	DDR_DQ[30]
R6	ATA_HIOW_N	E18	DDR_DQ[0]	A4	DDR_DQ[31]
Y3	ATA_INTRQ	D18	DDR_DQ[1]	A18	DDR_DQS[0]
W3	ATA_IORDY	C18	DDR_DQ[2]	D15	DDR_DQS[1]
AB2	ATA_RESET	B18	DDR_DQ[3]	A8	DDR_DQS[2]
W19	CINITRST_N	E17	DDR_DQ[4]	B5	DDR_DQS[3]
V19	CINT_N	D17	DDR_DQ[5]	B12	DDR_RAS_N
Y19	CSTSCHG	C17	DDR_DQ[6]	F11	DDR_VREF

Table 3–23: Cypher ESN7108A Alphabetical Pinout , continued

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
C12	DDR_WE_N	C22	EMI_CS_N[3]	G8	GND
AA5	EJ_TCK	F20	EMI_DQ[0]	G15	GND
Y5	EJ_TDI	F21	EMI_DQ[1]	H8	GND
V5	EJ_TDO	F22	EMI_DQ[2]	H10	GND
AB5	EJ_TMS	E20	EMI_DQ[3]	H11	GND
W5	EJ_TRST_N	E21	EMI_DQ[4]	H12	GND
L22	EMI_ADDR[0]	E22	EMI_DQ[5]	H13	GND
K17	EMI_ADDR[1]	D20	EMI_DQ[6]	H15	GND
K18	EMI_ADDR[2]	D21	EMI_DQ[7]	H21	GND
K19	EMI_ADDR[3]	B21	EMI_RD_DS_N	J8	GND
K20	EMI_ADDR[4]	B22	EMI_RDY	J10	GND
K21	EMI_ADDR[5]	B20	EMI_WR_N	J11	GND
K22	EMI_ADDR[6]	A2	GND	J12	GND
J17	EMI_ADDR[7]	A21	GND	J13	GND
J18	EMI_ADDR[8]	B3	GND	J15	GND
J19	EMI_ADDR[9]	B7	GND	K8	GND
J20	EMI_ADDR[10]	B10	GND	K10	GND
J21	EMI_ADDR[11]	B13	GND	K11	GND
J22	EMI_ADDR[12]	B16	GND	K12	GND
H17	EMI_ADDR[13]	B19	GND	K13	GND
H18	EMI_ADDR[14]	C3	GND	K15	GND
H19	EMI_ADDR[15]	C19	GND	L4	GND
H20	EMI_ADDR[16]	D4	GND	L8	GND
G17	EMI_ADDR[17]	D19	GND	L10	GND
G18	EMI_ADDR[18]	E4	GND	L11	GND
G19	EMI_ADDR[19]	E19	GND	L12	GND
G20	EMI_ADDR[20]	F6	GND	L13	GND
G21	EMI_ADDR[21]	F8	GND	L15	GND
G22	EMI_ADDR[22]	F12	GND	M8	GND
F19	EMI_ADDR[23]	F15	GND	M10	GND
D22	EMI_CS_N[0]	F17	GND	M11	GND
C20	EMI_CS_N[1]	F18	GND	M12	GND
C21	EMI_CS_N[2]	G2	GND	M13	GND

Table 3–23: Cypher ESN7108A Alphabetical Pinout , continued

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
M15	GND	H2	GPIO[3]	U18	HPI_WR_N
M19	GND	H3	GPIO[4]	F4	I2C_SCL
N8	GND	H4	GPIO[5]	F5	I2C_SDA
N10	GND	H5	GPIO[6]	K2	I2S_BCLK
N11	GND	H6	GPIO[7]	K3	I2S_LRCLK
N12	GND	P19	HPI_A[1]	K4	I2S_MCLK_IN
N13	GND	P20	HPI_A[2]	K6	I2S_MCLK_OUT
N15	GND	P21	HPI_A[3]	J2	I2S_SD
P8	GND	P22	HPI_A[4]	P6	MDC
P10	GND	N17	HPI_A[5]	N1	MDIO
P11	GND	N18	HPI_A[13]	M3	MII_COL
P12	GND	N19	HPI_A[14]	M4	MII_CRS
P13	GND	N20	HPI_A[15]	M5	MII_RXCLK
P15	GND	V20	HPI_CS_N	M6	MII_RXD[0]
R2	GND	U19	HPI_D[0]	L1	MII_RXD[1]
R8	GND	U20	HPI_D[1]	L2	MII_RXD[2]
R15	GND	U21	HPI_D[2]	L3	MII_RXD[3]
T8	GND	U22	HPI_D[3]	L6	MII_RXDV
T9	GND	T17	HPI_D[4]	L5	MII_RXER
T10	GND	T18	HPI_D[5]	M2	MII_TXCLK
T11	GND	T19	HPI_D[6]	N2	MII_TXD[0]
T12	GND	T20	HPI_D[7]	N3	MII_TXD[1]
T13	GND	R17	HPI_D[8]	N4	MII_TXD[2]
T14	GND	R18	HPI_D[9]	N5	MII_TXD[3]
T15	GND	R19	HPI_D[10]	N6	MII_TXEN
T21	GND	R20	HPI_D[11]	M1	MII_TXER
AA1	GND	R21	HPI_D[12]	B2	OSCPD
AA8	GND	R22	HPI_D[13]	Y18	PCI_AD[0]
AA15	GND	P17	HPI_D[14]	W18	PCI_AD[1]
AB21	GND	P18	HPI_D[15]	V18	PCI_AD[2]
J4	GPIO[0]	V21	HPI_INTC	AB17	PCI_AD[3]
J5	GPIO[1]	V22	HPI_INTD	AA17	PCI_AD[4]
J6	GPIO[2]	U17	HPI_RD_N	W17	PCI_AD[5]

Table 3–23: Cypher ESN7108A Alphabetical Pinout , continued

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
Y15	PCI_AD[6]	AB10	PCI_FRAME_N	L17	TSO_DQ[6]
AA14	PCI_AD[7]	V7	PCI_GNT_N[0]	L18	TSO_DQ[7]
Y14	PCI_AD[8]	V16	PCI_GNT_N[1]	N21	TSO_PSYNC
W14	PCI_AD[9]	Y16	PCI_GNT_N[2]	L21	TSO_TXCLK
V14	PCI_AD[10]	V17	PCI_GNT_N[3]	L19	TSO_TXCLKO
W13	PCI_AD[11]	W8	PCI_IDSEL	L20	TSO_VALID
V13	PCI_AD[12]	W6	PCI_INTA_N	P2	UART0_CTS
AB12	PCI_AD[13]	V6	PCI_INTB_N	P3	UART0_DCD
AA12	PCI_AD[14]	Y6	PCI_INTC_N	P1	UART0_DSR
W12	PCI_AD[15]	AA6	PCI_INTD_N	P4	UART0_DTR
AB11	PCI_AD[16]	Y12	PCI_IRDY_N	AB4	UART0_RI
AA11	PCI_AD[17]	AB14	PCI_PAR	W4	UART0_RTS
Y11	PCI_AD[18]	V15	PCI_PERR_N	Y4	UART0_SIN
W11	PCI_AD[19]	AA18	PCI_PME_N	AA4	UART1_SIN
V11	PCI_AD[20]	W7	PCI_REQ_N[0]	AB3	UART0_SOUT
W10	PCI_AD[21]	W16	PCI_REQ_N[1]	V4	UART1_SOUT
V10	PCI_AD[22]	AA16	PCI_REQ_N[2]	Y20	USB_STP
AB9	PCI_AD[23]	AB16	PCI_REQ_N[3]	W20	USB_CLK
AA9	PCI_AD[24]	AB6	PCI_RST_N	AA20	USB_DATA[0]
Y9	PCI_AD[25]	W15	PCI_SERR_N	AB20	USB_DATA[1]
W9	PCI_AD[26]	Y13	PCI_STOP_N	Y21	USB_DATA[2]
V9	PCI_AD[27]	AA10	PCI_TRDY_N	AA21	USB_DATA[3]
V8	PCI_AD[28]	K5	PLLBYPASS	Y22	USB_DATA[4]
AB7	PCI_AD[29]	F3	PLLRESET	AA22	USB_DATA[5]
AA7	PCI_AD[30]	P5	RST_N	W21	USB_DATA[6]
Y7	PCI_AD[31]	K1	SCAN_EN	W22	USB_DATA[7]
Y17	PCI_CBE[0]	J3	TEST_EN	AA19	USB_DIR
AB13	PCI_CBE[1]	N22	TSO_DQ[0]	AB19	USB_NEXT
Y10	PCI_CBE[2]	M17	TSO_DQ[1]	G9	VCORE
Y8	PCI_CBE[3]	M18	TSO_DQ[2]	G10	VCORE
V12	PCI_CLK	M20	TSO_DQ[3]	G11	VCORE
AB18	PCI_CLKRUN	M21	TSO_DQ[4]	G12	VCORE
AA13	PCI_DEVSEL_N	M22	TSO_DQ[5]	G13	VCORE

Table 3–23: Cypher ESN7108A Alphabetical Pinout , continued

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
G14	VCORE	G6	VID_PDATA[3]	N16	VIO33
H9	VCORE	F1	VID_PDATA[4]	P7	VIO33
H14	VCORE	F2	VID_PDATA[5]	P16	VIO33
J9	VCORE	E1	VID_PDATA[6]	R1	VIO33
J14	VCORE	E2	VID_PDATA[7]	R7	VIO33
K9	VCORE	E3	VID_PDATA[8]	R16	VIO33
K14	VCORE	D1	VID_PDATA[9]	T7	VIO33
L9	VCORE	B1	VID_VALID	T16	VIO33
L14	VCORE	C2	VID_VREF	T22	VIO33
M9	VCORE	A3	VIO25	U7	VIO33
M14	VCORE	A7	VIO25	U8	VIO33
N9	VCORE	A10	VIO25	U9	VIO33
N14	VCORE	A13	VIO25	U10	VIO33
P9	VCORE	A16	VIO25	U11	VIO33
P14	VCORE	A20	VIO25	U12	VIO33
R9	VCORE	E10	VIO25	U13	VIO33
R10	VCORE	E11	VIO25	U14	VIO33
R11	VCORE	E12	VIO25	U15	VIO33
R12	VCORE	E13	VIO25	U16	VIO33
R13	VCORE	A1	VIO33	AB1	VIO33
R14	VCORE	A22	VIO33	AB8	VIO33
F9	VDDA_DLL	G1	VIO33	AB15	VIO33
F14	VDDA_DLL	H7	VIO33	AB22	VIO33
G7	VDDA_DLL	H16	VIO33	F7	VSSA_DLL
G16	VDDA_DLL	H22	VIO33	F10	VSSA_DLL
M7	VDDA_PLL	J7	VIO33	F13	VSSA_DLL
D2	VID_FID	J16	VIO33	F16	VSSA_DLL
D3	VID_HREF	K7	VIO33	L7	VSSA_PLL
C1	VID_PCLK	K16	VIO33	H1	XIN
G3	VID_PDATA[0]	L16	VIO33	J1	XOUT
G4	VID_PDATA[1]	M16	VIO33		
G5	VID_PDATA[2]	N7	VIO33		

4. Datasheet History

1. Preliminary Data Sheet: “,Cypher ESN7108A” September 2006, 6251-695-1PD. First release of the Preliminary data sheet in the new format.
2. Preliminary Data Sheet: “Cypher ESN7108,” September 2006, 6251-695-2PD. Second release of the Preliminary data sheet. Major changes:

   - Micronas Cypher ESN7108 changed to Micronas Cypher ESN7108A.
3. Preliminary Data Sheet: “Cypher ESN7108A,” September 2007, 6251-695-3PD. Includes the following updates:

   - Updated 3.4. Mechanical Specifications section.
   - Updated the customer support URL address.

Micronas USA, Inc.
2805 Mission College Blvd., Santa Clara, CA 95054-1838, USA
Tel. +1-408-625-1200 · Fax +1-408-625-1299 · http://service.micronas.com · Internet: www.micronasusa.com