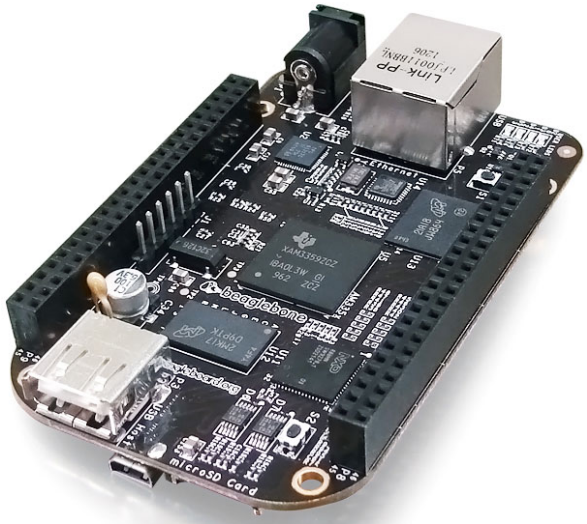

# BeagleBone Black System Reference Manual

**Revision A4**
**January 2, 2013**

**Send all comments and errors concerning this document to the author at**
**gerald@beagleboard.org**

**For other questions you may contact:**
**Gerald Coley**
**Texas Instruments**
**12500 TI Blvd. Dallas, Tx 75243**
**g-coley1@ti.com**

***PRELIMINARY NOT FOR DISCLOSURE***

# Table of Contents

**FIGURES** ..... **4**

**TABLES** ..... **5**

**1.0 INTRODUCTION** ..... **6**

**2.0 CHANGE HISTORY** ..... **6**

2.1 CHANGE HISTORY ..... 6

**3.0 BEAGLEBONE BLACK OVERVIEW** ..... **6**

3.1 BEAGLEBONE COMPATIBILITY ..... 6
3.2 IN THE BOX ..... 7
3.3 SERIAL DEBUG CABLE ..... 8
3.4 HDMI CABLE ..... 8
3.5 5VDC POWER SUPPLY ..... 9

**4.0 BEAGLEBONE BLACK FEATURES AND SPECIFICATION** ..... **10**

4.1 BOARD COMPONENT LOCATIONS ..... 11

**5.0 BEAGLEBONE BLACK HIGH LEVEL SPECIFICATION** ..... **12**

5.1 BLOCK DIAGRAM ..... 12
5.2 PROCESSOR ..... 12
5.3 MEMORY ..... 13
*5.3.1 DDR3L ..... 13*
*5.3.2 EEPROM ..... 13*
*5.3.3 Embedded MMC ..... 13*
*5.3.4 MicroSD Connector ..... 13*
*5.3.5 Boot Modes ..... 14*
5.4 POWER MANAGEMENT ..... 14
5.5 PC USB INTERFACE ..... 15
5.6 SERIAL DEBUG PORT ..... 15
5.7 USB1 HOST PORT ..... 15
5.8 POWER SOURCES ..... 15
5.9 RESET BUTTON ..... 16
5.10 INDICATORS ..... 16
5.11 CTI JTAG HEADER ..... 16
5.12 HDMI INTERFACE ..... 16
5.13 CAPE BOARD SUPPORT ..... 16

**6.0 DETAILED HARDWARE DESIGN** ..... **18**

6.1 POWER SECTION ..... 19
*6.1.1 TPS65217C PMIC ..... 19*
*6.1.2 DC Input ..... 21*
*6.1.3 USB Power ..... 21*
*6.1.4 Power Selection ..... 22*
*6.1.5 Power Consumption ..... 23*
*6.1.6 Processor Interfaces ..... 23*
*6.1.7 Power Rails ..... 24*
*6.1.8 Power LED ..... 27*
*6.1.9 TPS65217C Power Up Process ..... 27*
*6.1.10 Processor Control Interface ..... 28*
*6.1.11 Low Power Mode Support ..... 28*
6.2 XAM3359ZCZ PROCESSOR ..... 29
*6.2.1 Description ..... 29*

*6.2.2 High Level Features* ..... *30*
*6.2.3 Documentation* ..... *30*
6.3 DDR3 MEMORY ..... 31
*6.3.1 Memory Device* ..... *31*
*6.3.2 DDR3 Memory Design* ..... *31*
*6.3.3 Power Rails* ..... *33*
*6.3.4 VREF* ..... *33*
6.4 EMMC MEMORY ..... 34
*6.4.1 eMMC Device* ..... *34*
*6.4.2 eMMC Circuit Design* ..... *34*
6.5 MICRO SECURE DIGITAL ..... 36
*6.5.1 uSD Design* ..... *36*
6.6 USER LEDS ..... 37
6.7 BOOT CONFIGURATION ..... 37
*6.7.1 Boot Configuration Design* ..... *37*
*6.7.2 Boot Options* ..... *38*
6.8 10/100 ETHERNET ..... 39
*6.8.1 Ethernet Processor Interface* ..... *39*
*6.8.2 Ethernet Connector Interface* ..... *40*
*6.8.3 Ethernet PHY Power, Reset, and Clocks* ..... *41*
*6.8.4 LAN8710A Mode Pins* ..... *42*
6.9 HDMI INTERFACE ..... 42
*6.9.1 HDMI Framer* ..... *43*
*6.9.2 HDMI Video Processor Interface* ..... *43*
*6.9.3 HDMI Control Processor Interface* ..... *43*
*6.9.4 Interrupt Signal* ..... *44*
*6.9.5 Audio Interface* ..... *44*
*6.9.6 Power Connections* ..... *45*
*6.9.7 HDMI Connector Interface* ..... *46*

**7.0 CONNECTORS ..... 47**

7.1 EXPANSION CONNECTORS ..... 47
*7.1.1 Connector P8* ..... *48*
*7.1.2 Connector P9* ..... *50*
7.2 POWER JACK ..... 52
7.3 USB CLIENT ..... 53
7.4 USB HOST ..... 54
7.5 SERIAL HEADER ..... 55
7.6 HDMI ..... 56
7.7 USD ..... 57
7.8 ETHERNET ..... 58

**8.0 CAPE BOARD SUPPORT ..... 59**

8.1 EEPROM ..... 59
*8.1.1 EEPROM Address* ..... *60*
*8.1.2 I2C Bus* ..... *60*
*8.1.3 EEPROM Write Protect* ..... *61*
*8.1.4 EEPROM Data Format* ..... *62*
*8.1.5 Pin Usage* ..... *63*
8.2 PIN USAGE CONSIDERATION ..... 67
*8.2.1 Boot Pins* ..... *67*
8.3 EXPANSION CONNECTORS ..... 68
*8.3.1 Non-Stacking Headers-Single Cape* ..... *68*
*8.3.2 Battery Connector- Single* ..... *69*
*8.3.3 Main Expansion Headers-Stacking* ..... *70*

*8.3.4 Stacked Capes w/Signal Stealing* ..... *71*
*8.3.5 Retention Force* ..... *71*
*8.3.6 BeagleBone Black Female Connectors* ..... *72*
8.4 SIGNAL USAGE ..... 72
8.5 CAPE POWER ..... 73
*8.5.1 Main Board Power* ..... *73*
*8.5.2 Expansion Board External Power* ..... *74*
8.6 MECHANICAL ..... 74
*8.6.1 Standard Cape Size* ..... *75*
*8.6.2 Extended Cape Size* ..... *75*
*8.6.3 Enclosures* ..... *76*
**9.0 BEAGLEBONE BLACK MECHANICAL SPECIFICATION ..... 77**
**10.0 DESIGN INFORMATION ..... 79**

# Figures

Figure 1. FTDI Serial cable ..... 8
Figure 2. Micro HDMI Cable ..... 8
Figure 3. 5VDC Power Supply ..... 9
Figure 4. Key Components ..... 11
Figure 5. BeagleBone Black Block Diagram ..... 12
Figure 6. Block Diagram ..... 18
Figure 7. High Level Power Block Diagram ..... 19
Figure 8. TPS65217C Block Diagram ..... 20
Figure 9. DC Power Connections ..... 21
Figure 10. USB Power Connections ..... 22
Figure 11. Power Rails ..... 24
Figure 12. Power Sequencing ..... 26
Figure 13. Power Sequencing ..... 26
Figure 14. Power Processor Interfaces ..... 27
Figure 15. XAM3359 Block Diagram ..... 29
Figure 16. DDR3 Memory Design ..... 32
Figure 17. DDR3 VREF Design ..... 33
Figure 18. eMMC Memory Design ..... 35
Figure 19. uSD Design ..... 36
Figure 20. User LEDs ..... 37
Figure 21. Processor Boot Configuration Design ..... 38
Figure 22. Processor Boot Configuration ..... 38
Figure 23. Ethernet Processor Interface ..... 39
Figure 24. Ethernet Connector Interface ..... 40
Figure 25. Ethernet PHY, Power, Reset, and Clocks ..... 41
Figure 26. Ethernet PHY Mode Pins ..... 42
Figure 27. HDMI Framer Processor Interface ..... 44
Figure 28. HDMI Power Connections ..... 45
Figure 29. Connector Interface Circuitry ..... 46
Figure 30. Expansion Connector Location ..... 47

Figure 31. 5VDC Power Jack ..... 52
Figure 32. USB Client Connector ..... 53
Figure 33. USB Host Connector ..... 54
Figure 34. Serial Debug Header ..... 55
Figure 35. FTDI USB to Serial Adapter ..... 55
Figure 36. HDMI Connector ..... 56
Figure 37. HDMI Connector ..... 56
Figure 38. uSD Connector ..... 57
Figure 39. Ethernet Connector ..... 58
Figure 40. Expansion Board EEPROM No Write Protect ..... 60
Figure 41. Expansion Board EEPROM Write Protect ..... 61
Figure 42. Expansion Boot Pins ..... 67
Figure 43. Single Expansion Connector ..... 68
Figure 44. Single Cape Expansion Connector ..... 69
Figure 45. Battery/Backlight Expansion Connector ..... 69
Figure 46. Expansion Connector ..... 70
Figure 47. Stacked Cape Expansion Connector ..... 70
Figure 48. Stacked w/Signal Stealing Expansion Connector ..... 71
Figure 49. Connector Pin Insertion Depth ..... 72
Figure 50. Cape Board Dimensions ..... 75
Figure 51. Board Top Side Profile ..... 77
Figure 52. Board Bottom Profile ..... 78

## Tables

Table 1. Change History ..... 6
Table 2. BeagleBone Black Features ..... 10
Table 3. BeagleBone Power Consumption(mA@5V) ..... 23
Table 4. Processor Features ..... 30
Table 5. eMMC Boot Pins ..... 35
Table 6. TDA19988 I2C Address ..... 43
Table 7. Expansion Header P8 Pinout ..... 49
Table 8. Expansion Header P9 Pinout ..... 51
Table 9. Expansion Board EEPROM ..... 62
Table 10. EEPROM Pin Usage ..... 64
Table 11. Single Cape Connectors ..... 69
Table 12. Single Cape Backlight Connectors ..... 70
Table 13. Stacked Cape Connectors ..... 71
Table 14. Expansion Voltages ..... 73

# 1.0 Introduction

This document is the **System reference Manual** for the BeagleBone Black. It covers the design for the BeagleBone Black. The board will be referred to in the remainder of this document as BeagleBone Black. There are also references to the original BeagleBone as well.

This design is subject to change without notice as we will work to keep improving the design as the product matures.

# 2.0 Change History

## 2.1 Change History

**Table 1. Change History**

| Rev | Changes | Date | By |
|---|---|---|---|
| A4 | Preliminary | January 4, 2013 | GC |
| | | | |
| | | | |
| | | | |
| | | | |

# 3.0 BeagleBone Black Overview

The BeagleBone Black is the latest addition to the BeagleBoard.org family and like its' predecessors, is designed to address the Open Source Community, early adopters, and anyone interested in a low cost ARM Cortex A8 based processor. It has been equipped with a minimum set of features to allow the user to experience the power of the processor and is not intended as a full development platform as many of the features and interfaces supplied by the processor are not accessible from the BeagleBone Black via onboard support of some interfaces.

## 3.1 BeagleBone Compatibility

The BeagleBone Black is intended to be compatible with the original BeagleBone as much as possible. There are a several areas where there are differences between the two designs. These differences are listed below along with the reasons for the differences.

- AM3358A Processor, 800MHz operation
- 512MB DDR3L
    - Cost reduction

  - Performance increase
  - Memory size increase
  - Lower power
- No Serial port by default.
  - Cost reduction
  - Can be added by buying a TTL to USB Cable that is widely available
- No JTAG emulation over USB.
  - Cost reduction
  - JTAG header is not populated, but can easily be mounted.
- Onboard Managed NAND
  - Cost reduction
  - Performance boost x8 vs. x4 bits
  - Performance boost due to deterministic properties vs. SD card
- GPMC bus may not be accessible from the expansion headers in some cases
  - Result of eMMC on the main board
  - Signals are routed to the expansion connector
  - If eMMC is not used, signals can be used via expansion if eMMC is held in reset
- There may be 10 less GPIO pins available
  - Result of eMMC
  - If eMMC is not used, could be used
- No power expansion Header
  - Cost reduction
  - Space reduction
- HDMI interface onboard
  - Feature addition
  - Audio and video capable
  - Micro HDMI
- No onboard USB JTAG emulation
  - Major cost reduction
- No three function USB cable
  - Major cost reduction

## 3.2 In The Box

The BeagleBone Black will ship with the following components:

- BeagleBone Black
- 5 pin miniUSB Cable

### 3.3 Serial Debug Cable

Additional cables that are not supplied with the board may be needed.

To access the serial debug port on the processor, a serial to TTL cable is required. The part number is TTL-232R-3V3 and can be purchased from numerous different sources.

This cable can be purchased for FTDI at
http://apple.clickandbuild.com/cnb/shop/ftdichip?op=catalogue-products-null&prodCategoryID=105&title=USB-TTL+0.1%94+Socket

For a list of sales channels go to http://www.ftdichip.com/FTSalesNetwork.htm

**Figure 1. FTDI Serial cable**

### 3.4 HDMI Cable

To access the HDMI output, a microHDMI cable is required as pictured below.

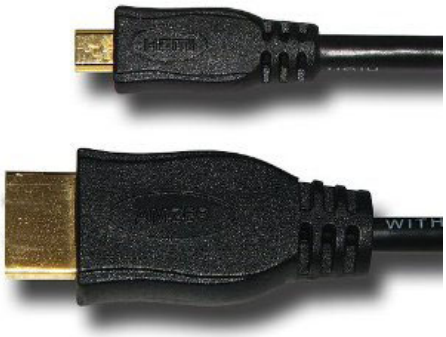

**Figure 2. Micro HDMI Cable**

The cable is available from numerous sources such as Amazon
http://www.amazon.com/Amzer-Micro-HDMI-Speed-Cable/dp/B003OBZSHC.

Prices can range from $10 to $25.

### 3.5 5VDC Power Supply

Current via the USB port is limited to 500mA by the power management device on the board. Exceeding this current will cause the board to shut off. Running from an external DC power supply solves this issue. The board uses the same power supply as the original BeagleBoard. A minimum of 5V at 1A is recommended. The power supply should be well regulated and 5V +/-.5V.

A good choice for a power supply can be found at http://www.adafruit.com/products/276

**Figure 3. 5VDC Power Supply**

## 4.0 BeagleBone Black Features and Specification

This section covers the specifications and features of the BeagleBone Black and provides a high level description of the major components and interfaces that make up the board.

**Table 2** provides a list of the BeagleBone Black features.

**Table 2. BeagleBone Black Features**

| | Feature | |
|---|---|---|
| **Processor** | AM3358/9<br>600MHZ-USB Powered (TBD)<br>800MHZ-DC Powered | |
| **SDRAM Memory** | 512MB DDR3L 606MHZ | |
| **Flash eMMC** | 2GB, 8bit | |
| **PMIC TPS65217C** | PMIC regulator and one additional LDO. | |
| **Debug Support** | Optional Onboard 20-pin CTI JTAG | |
| **Power** | miniUSB USB or DC Jack | 5VDC External Via Expansion Header |
| **PCB** | 3.4" x 2.1" | 6 layers |
| **Indicators** | 1-Power, 2-Ethernet, 4-User Controllable LEDs | |
| **HS USB 2.0 Client Port** | Access to the USB1 Client mode via miniUSB | |
| **HS USB 2.0 Host Port** | USB Type A Socket, 500mA LS/FS/HS | |
| **Serial Port** | UART0 access via 6 pin Header. Header is populated | |
| **Ethernet** | 10/100, RJ45 | |
| **SD/MMC Connector** | microSD , 3.3V | |
| **User Input** | 1-Reset Button, 1-User Boot Button | |
| **Video Out** | 16b HDMI , w/ CEC | |
| **Audio** | Via HDMI Interface | |
| **Expansion Connectors** | Power 5V, 3.3V , VDD_ADC(1.8V)<br>3.3V I/O on all signals<br>McASP0, SPI1, I2C, GPIO(65), LCD, GPMC, MMC1, MMC2, 7 AIN(1.8V MAX), 4 Timers, 3 Serial Ports, CAN0, EHRPWM(0,2),XDMA Interrupt, Power button, Expansion Board ID<br>(Up to 4 can be stacked) | |
| **Weight** | 1.4 oz (39.68 grams) | |

***NOTE: THE INITIAL A4 VERSIONS WERE BUILT USING THE AM3352 PROCESSOR. THIS WAS A RESULT OF RECEIVING MISMARKED PARTS FROM THE SUPPLIER. DUE TO THE TIGHT SCHEDULE, THE DECSION WAS MADE TO BUILD WITH THE AM3352 VERSION AS REV A4. PRODUCTION VERSION IS REV A5 AND WILL HAVE THE CORRECT PROCESSOR. REV A4 DOES NOT HAVE SUPPORT FOR THE PRU OR SGX***

## 4.1 Board Component Locations

**Figure 4** below shows the locations of the key components on the PCB layout of the BeagleBone Black.

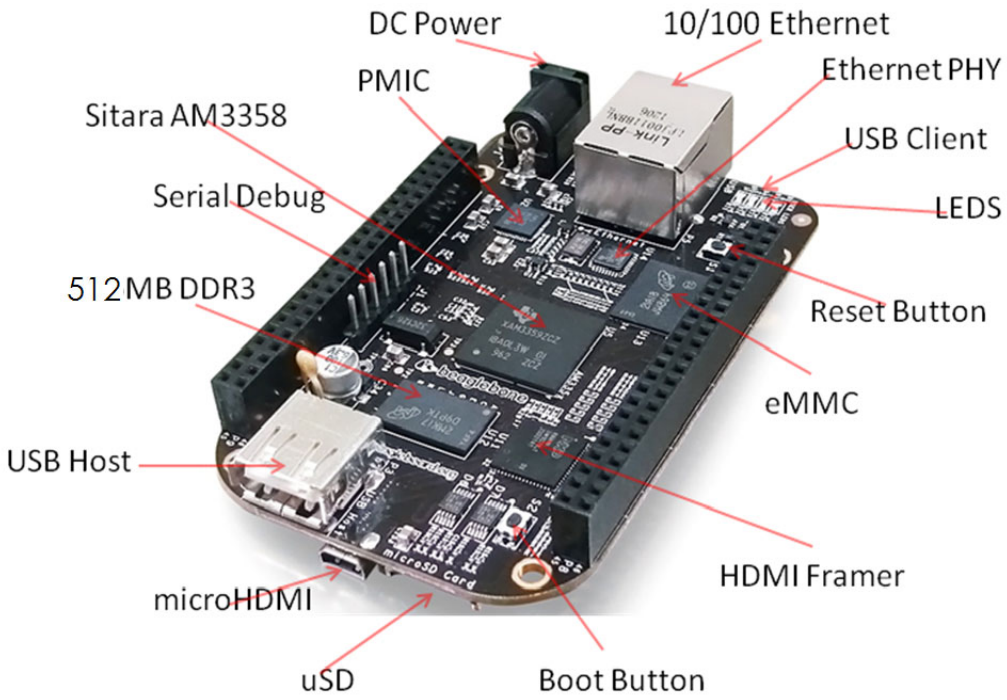


**Figure 4. Key Components**

The **Sitara AM3358** is the processor.
**512MB DDR3** is the processor dynamic RAM memory.
**Serial Debug** is the serial debug port.
**PMIC** provides the power rails to the various components on the board.
**DC Power** is the main DC input that accepts 5V power.
**10/100 Ethernet** is the connection to the LAN.
**Ethernet PHY** is the physical interface to the network.
**USB Client** is a miniUSB connection to a PC that can also power the board.
There are four blue **LED**S that can be used by the user.
**Reset Button** allows the user to reset the processor.
**eMMC** is an onboard MMC chip that hold sup to 2GB of data.
**HDMI** Framer provides control for an HDMI or DVI-D display.
**BOOT Button** can be used to force a boot from the SD card or from the USB port.
**uSD** slot is where a uSD card can be installed.
The **microHDMI** connector is where the display is connected.
**USB Host** can be connected different USB interfaces such as Wifi, BT, Keyboard, etc,

# 5.0 BeagleBone Black High Level Specification

This section provides the high level specification of the BeagleBone Black.

## 5.1 Block Diagram

**Figure 5** below is the high level block diagram of the BeagleBone Black.

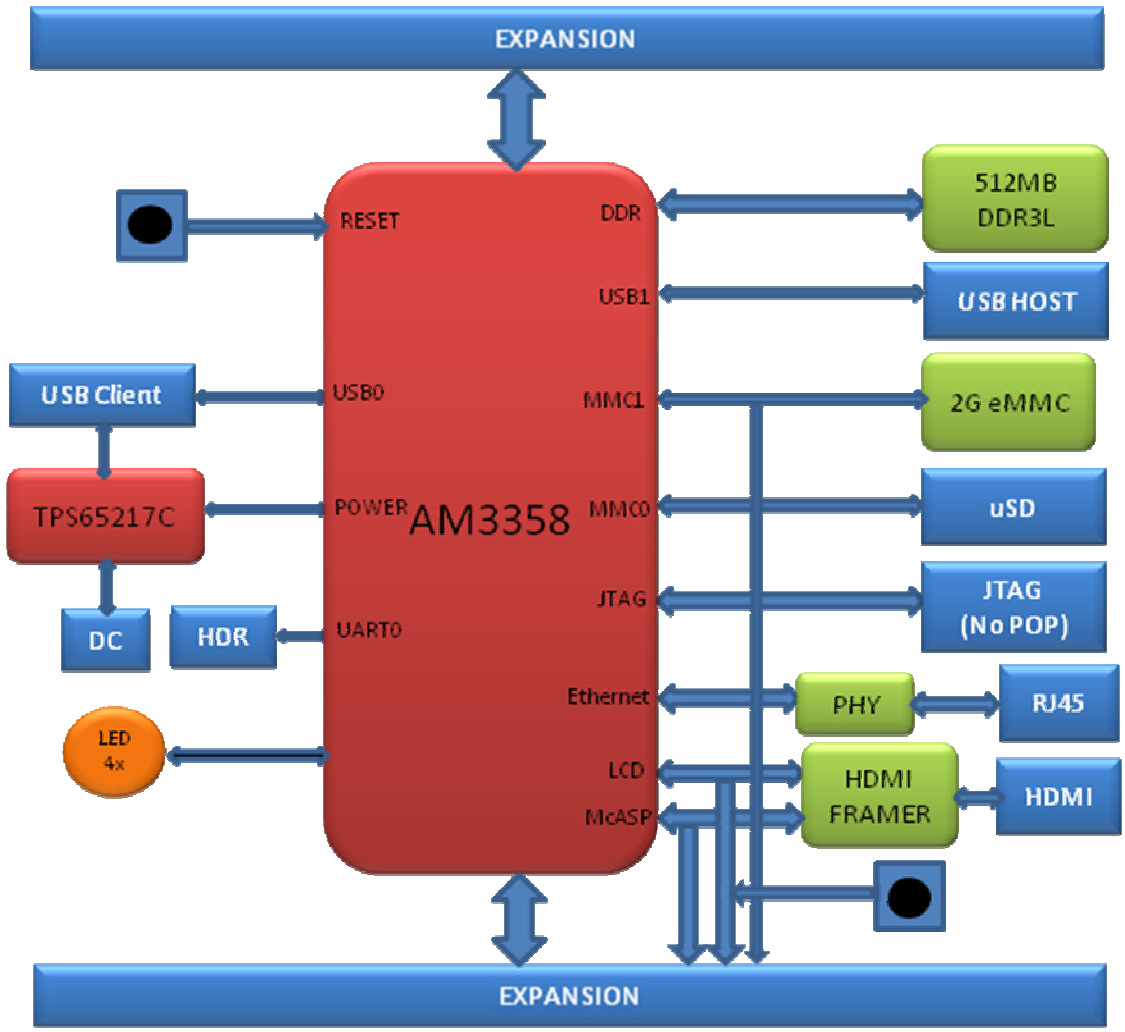


**Figure 5. BeagleBone Black Block Diagram**

## 5.2 Processor

For the initial release, the board uses the XAM3359AZCZ processor in the 15x15 package. This is the same processor as used on the original BeagleBone. It does use the updated 2.0 revision with several fixes as opposed to the original BeagleBone. None of these fixes provide substantial additional features. Eventually the board will move to the AM3358AZCZ device once readily available. The move does provide for a frequency increase to 800MHz.

***NOTE: THE INITIAL A4 VERSIONS WERE BUILT USING THE AM3352 PROCESSOR. THIS WAS A RESULT OF RECEIVING MISMARKED PARTS FROM THE SUPPLIER. DUE TO THE TIGHT SCHEDULE, THE DECSION WAS MADE TO BUILD WITH THE AM3352 VERSION AS REV A4. PRODUCTION VERSION IS REV A5 AND WILL HAVE THE CORRECT PROCESSOR. REV A4 DOES NOT HAVE SUPPORT FOR THE PRU OR SGX***

## 5.3 Memory

Described in the following sections are the three memory devices found on the BeagleBone Black.

### 5.3.1 DDR3L

A single 512Mb x16 bit DDR3L 4Gb memory device is used. The memory used is the MT41K512M16HA-125 from Micron. It will operate at a clock frequency of 303MHz yielding an effective rate of 606MHZ on the DDR3 bus allowing for 1.2GB of DDR3 bandwidth.

### 5.3.2 EEPROM

A single 32KB EEPROM is provided on I2C0 that holds the board information. This information includes board name, serial number, and revision information. This will be the same as found on the original BeagleBone. It has a test point to allow the device to be programmed and otherwise to provide write protection when not grounded.

### 5.3.3 Embedded MMC

A single 2GB embedded MMC (eMMC) device is on the board. The device will connect to the MMC1 port of the processor, allowing for 8bit wide access. Default boot mode for the board will be MMC1 with an option to change it to MMC0 for SD card booting. MMC0 cannot be used in 8Bit mode because the lower data pins are located on the pins used by the Ethernet port. But this does not interfere with SD card operation but it does make it unsuitable for use as an eMMC port if the 8 bit feature is needed.

### 5.3.4 MicroSD Connector

The board is equipped with a single microSD connector to act as the secondary boot source for the board and if selected as such, can be the primary boot source. The connector will support larger capacity SD cards. No SD card is provided with the board. Booting from MMC0 will be used to flash the eMMC in the production environment or by the user to update the SW as needed.

### 5.3.5 Boot Modes

As mentioned earlier, there are four boot modes supported:

- **eMMC Boot…**This is the default boot mode and will allow for the fastest boot time and will enable the board to boot out of the box without having to purchase an SD card or an SD card writer.
- **SD Boot…**This mode will boot from the uSD slot. This mode can be used to override what is on the eMMC device and can be used to program the eMMC when used in the manufacturing process or for filed updates.
- **Serial Boot…**This mode will use the serial port to allow downloading of the software direct. A separate serial cable is required to use this port.
- **USB Boot…**This mode supports booting over the USB port.

A switch is provided to allow switching between the modes.

- ❖ Holding the switch down during boot without an SD card will force the boot source to be the USB port and if nothing is detected on the USB port, it will go to the serial port for download.
- ❖ Without holding the switch, the board will boot from eMMC. If it is empty, then it will try booting from the uSD slot, followed by the serial port, and then the USB port.

## 5.4 Power Management

The **TPS65127C** power management device is used along with a separate LDO to provide power to the system. The **TPS65127C** version provides for the proper voltages required for DDR3. This is the same device as used on the original BeagleBone with the exception of the power rail configuration settings which will be changed in the internal EEPROM to the TPS65217 to support the new voltages.

DDR3 requires 1.5V instead of 1.8V on the DDR2 as is the case on the original BeagleBone. The 1.8V regulator has been changed to 1.5V for the DDR3. The LDO3 3.3V rail has been changed to 1.8V to support those rails on the processor. LDO4 is still 3.3V for the 3.3V rails on the processor. An external **LDOTLV70233** provides the 3.3V rail for the rest of the board.

## 5.5 PC USB Interface

The board has a miniUSB connector that connects the USB0 port to the processor. This is the same connector as used on the original BeagleBone.

## 5.6 Serial Debug Port

Serial debug is provided via UART0 on the processor via a single 1x6 pin header. In order to use the interface a USB to TTL adapter will be required. The header is compatible with the one provided by FTDI and can be purchased for about $12 to $20 from various sources. Signals supported are TX and RX. None of the handshake signals are supported.

## 5.7 USB1 Host Port

On the board is a single USB Type A female connector with full LS/FS/HS Host support that connects to USB1 on the processor. The port can provide power on/off control and up to 500mA of current at 5V. Under USB power, the board will not be able to supply the full 500mA, but should be sufficient to supply enough current for a lower power USB device supplying power between 50 to 100mA.

You can use a wireless keyboard/mouse configuration or you can add a HUB for standard keyboard and mouse interfacing.

## 5.8 Power Sources

The board can be powered from four different sources:

- A USB port on a PC
- A 5VDC 1A power supply plugged into the DC connector.
- A power supply with a USB connector.
- Expansion connectors

The USB cable is shipped with each board. This port is limited to 500mA by the Power Management IC.

The power supply is not provided with the board but can be easily obtained from numerous sources. A 1A supply is sufficient to power the board, but if there is a cape plugged into he board, then more current may needed from the DC supply.

Power routed to the board via the expansion header could be provided from power derived on a cape.

The DC supply should be well regulated and 5V +/-.25V.

### 5.9 Reset Button

When pressed and released, causes a reset of the board. The reset button used on the BeagleBone Black is a little larger than the one used on the original BeagleBone. It has also been moved out to the edge of the board so that it is more accessible.

### 5.10 Indicators

There are five total blue LEDs on the board.

- One blue power LED indicates that power is applied and the power management IC is up. If this LED flashes when applying power, it means that an excess current flow was detected and the PMIC has shutdown.
- Four blue LEDs that can be controlled via the SW by setting GPIO pins.

In addition, there are two LEDs on the RJ45 to provide Ethernet status indication. One is yellow and the other is green.

### 5.11 CTI JTAG Header

A place for an optional 20 pin CTI JTAG header is provided on the board to facilitate the SW development and debugging of the board by using various JTAG emulators. This header is not supplied standard on the board. To use this, a connector will need to be soldered onto the board.

### 5.12 HDMI Interface

A single HDMI interface is connected to the 16pin LCD interface on the processor. The NXP TDA19988BHN is used to convert the LCD interface to HDMI and convert the audio as well. The HDMI device does not support HDCP copy protection.

The signals are still connected to the expansion headers to enable the use of LCD expansion boards or access to other functions on the board as needed.

### 5.13 Cape Board Support

The BeagleBone Black has the ability to accept up to four expansion boards or capes that can be stacked onto the expansion headers. The word cape comes from the shape of the board as it is fitted around the Ethernet connector on the main board. This notch acts as a key to insure proper orientation of the Cape.

The majority of capes designed for the original BeagleBone will work on the BeagleBone Black. The two main expansion headers will be populated on the board. There are a few

exceptions where certain capabilities may not be present or are limited to the BeagleBone Black. These include:

- GPMC bus may NOT be available due to the use of those signals by the eMMC. If the eMMC is used for booting only and the file system is on the SD card, then these signals could be used.
- Another option is to use the SD or serial boot modes and not use the eMMC.
- The power expansion header is not on the BeagleBone Black so those functions are not supported.

## 6.0 Detailed Hardware Design

**Figure 6** below is the high level block diagram of the BeagleBone Black.

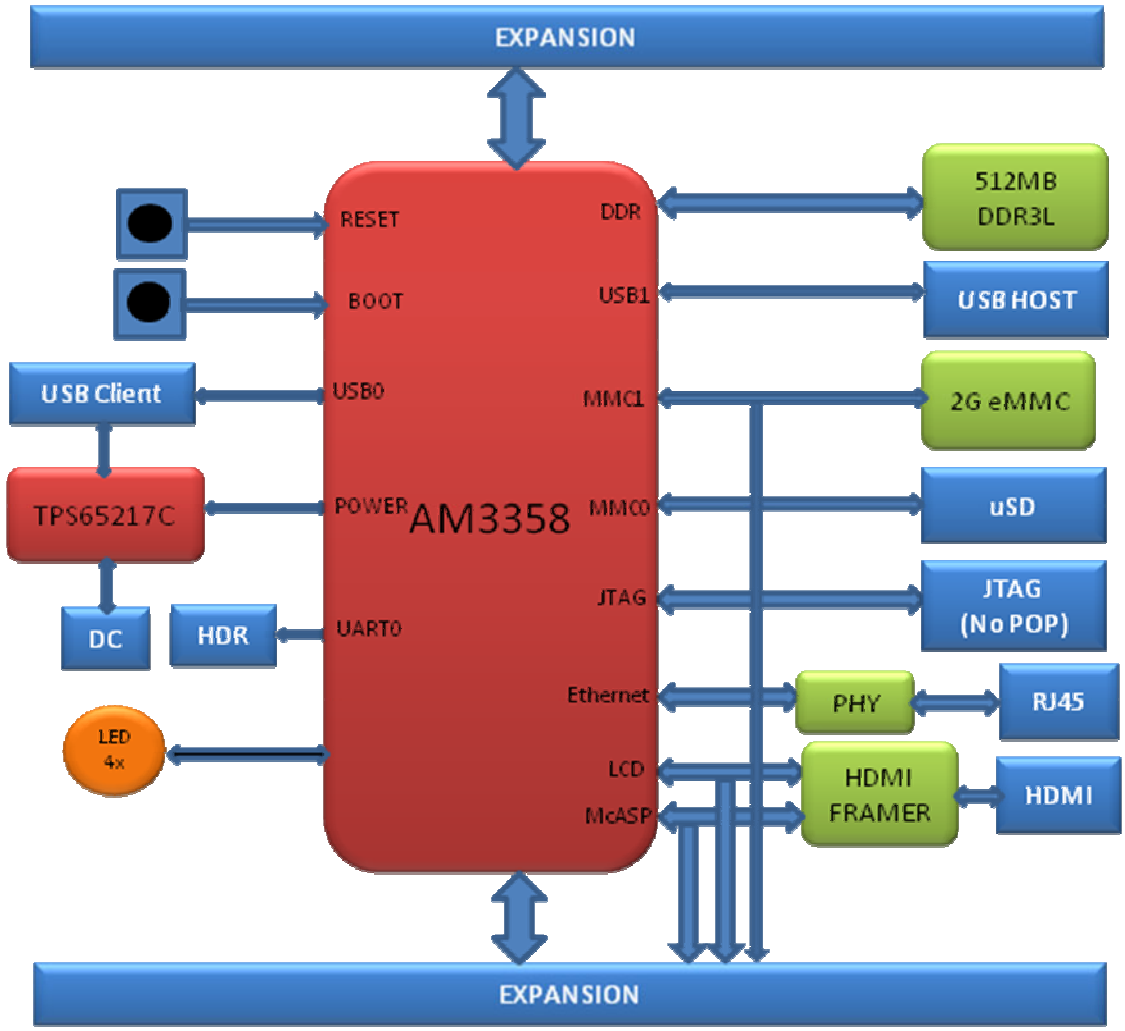


**Figure 6. Block Diagram**

## 6.1 Power Section

**Figure 7** is the high level block diagram of the power section.

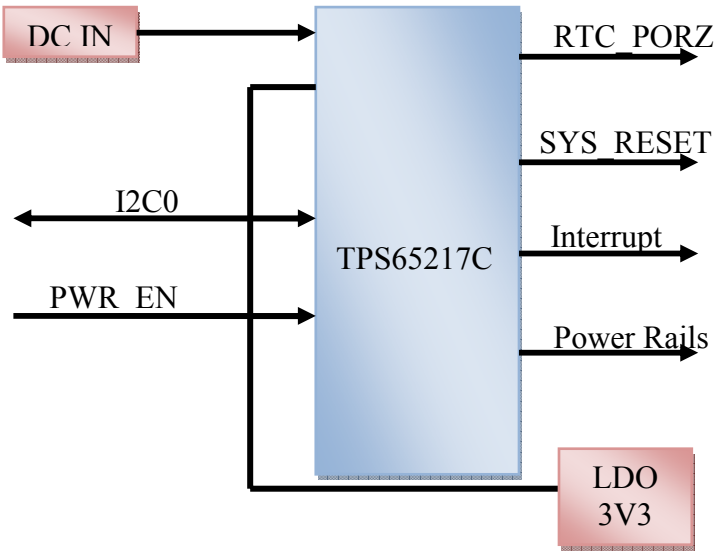


**Figure 7. High Level Power Block Diagram**

This section describes the power section of the design and all the functions performed by the **TPS65217C**.

### 6.1.1 TPS65217C PMIC

The main Power Management IC (PMIC) in the system is the **TPS65217C** which is a single chip power management IC consisting of a linear dual-input power path, three step-down converters, and four LDOs. The system is supplied by a USB port or DC adapter. Three high-efficiency 2.25MHz step-down converters are targeted at providing the core voltage, MPU, and memory voltage for the board.

The step-down converters enter a low power mode at light load for maximum efficiency across the widest possible range of load currents. For low-noise applications the devices can be forced into fixed frequency PWM using the I2C interface. The step-down converters allow the use of small inductors and capacitors to achieve a small solution size.

LDO1 and LDO2 are intended to support system-standby mode. In normal operation they can support up to 100mA each. LDO3 and LDO4 can support up to 285mA each.

By default only LDO1 is always ON but any rail can be configured to remain up in SLEEP state. Especially the DCDC converters can remain up in a low-power PFM mode to support processor suspend mode. The **TPS65217C** offers flexible power-up and

power-down sequencing and several house-keeping functions such as power-good output, pushbutton monitor, hardware reset function and temperature sensor to protect the battery. Note that support for the battery is not provided on the BeagleBone Black

For more information on the **TPS65217C**, refer to http://www.ti.com/product/tps65217 .

**Figure 8** is the high level block diagram of the **TPS65217C**.

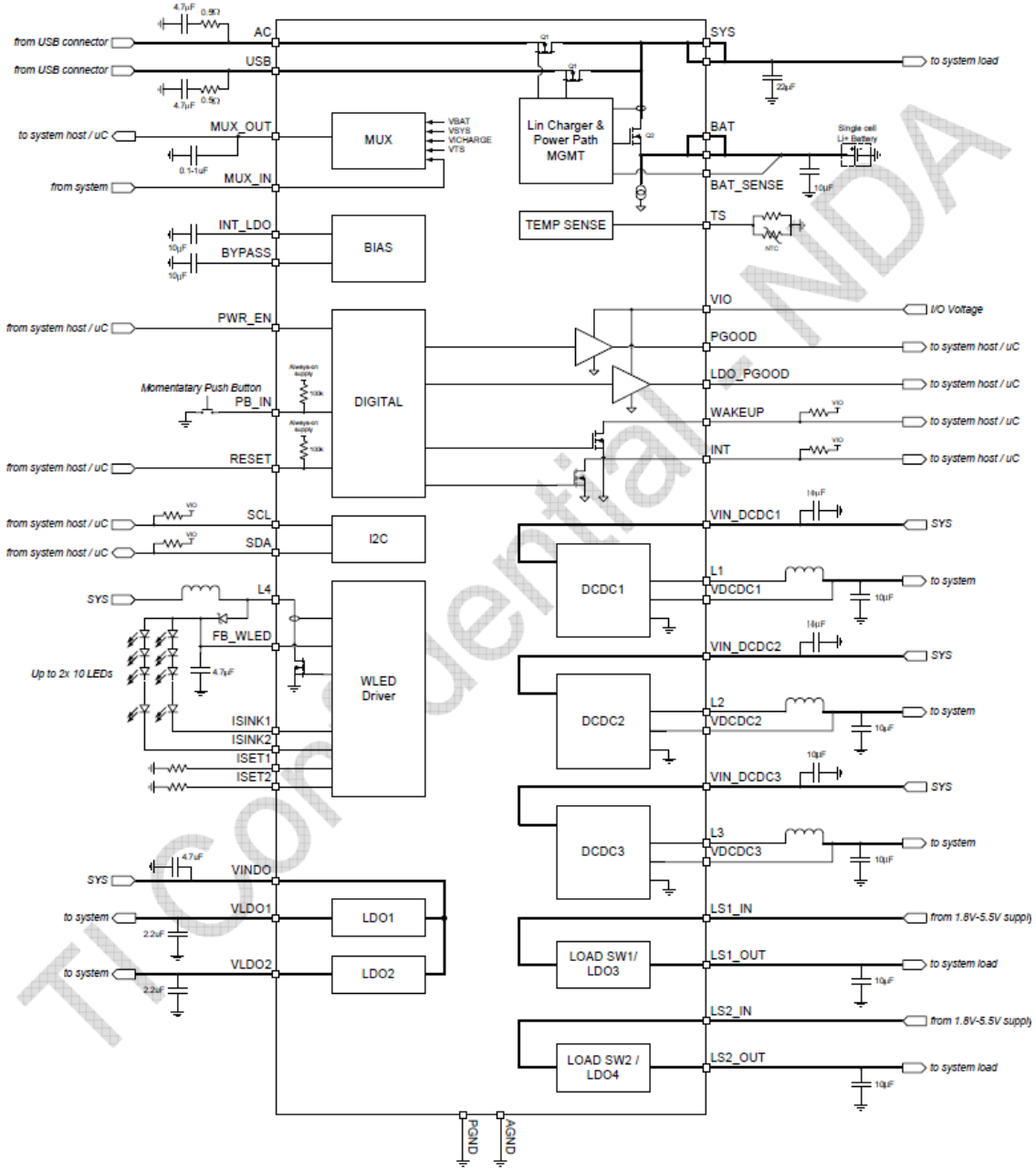


**Figure 8. TPS65217C Block Diagram**

### 6.1.2 DC Input

The **Figure 9** shows how the DC input is connected to the TPS65217C.

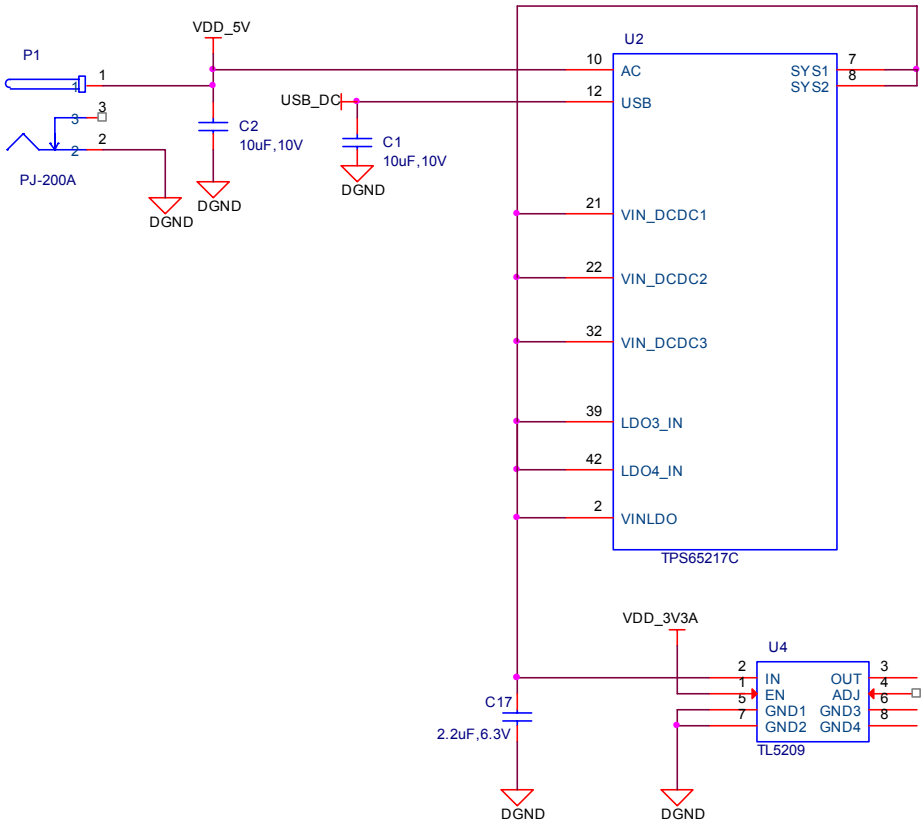


**Figure 9. DC Power Connections**

A 5VDC supply can be used to provide power to the board. The power supply current depends on how many and what type of add on boards are connected to the board. For typical use, a 5VDC supply rated at 1A should be sufficient. If heavier use of the expansion headers or USB host port is expected, then a higher current supply will be required.

The connector used is a 2.1MM center positive x 5.5mm outer barrel. The 5VDC rail is connected to the expansion header. It is possible to power the board via the expansion headers from an add-on card. The 5VDC is also available for use by the add-on cards when the power is supplied by the 5VDC jack on the board.

### 6.1.3 USB Power

The board can also be powered from the USB port. A typical USB port is limited to 500mA max. When powering from the USB port, the VDD_5V rail is not provided to

the expansion header. So capes that require the 5V rail to supply the cape direct, bypassing the **TPS65217C**, will not have that rail available for use. The 5VDC supply from the USB port is provided on the SYS_5V, the one that comes from the **TPS65217C**, rail of the expansion header for use by a cape. **Figure 10** is the design of the USB power input section.

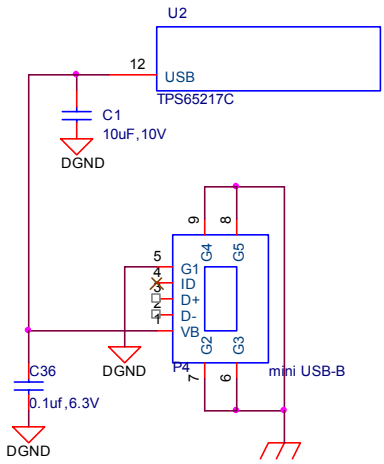


**Figure 10. USB Power Connections**

### 6.1.4 Power Selection

The selection of either the 5VDC or the USB as the power source is handled internally to the **TPS65217C** and automatically switches to 5VDC power if both are connected. SW can change the power configuration via the I2C interface from the processor. In addition, the SW can read the **TPS65217C** and determine if the board is running on the 5VDC input or the USB input. This can be beneficial to know the capability of the board to supply current for things like operating frequency and expansion cards.

It is possible to power the board from the USB input and then connect the DC power supply. The board will switch over automatically to the DC input.

### 6.1.5 Power Consumption

The power consumption of the board varies based on power scenarios and the board boot processes. **Table 3** is an analysis of the power consumption of the board in these various scenarios.

**Table 3. BeagleBone Black Power Consumption(mA@5V)**

| MODE | USB | DC | DC+USB |
|---|---|---|---|
| **Reset** | TBD | TBD | TBD |
| **UBoot** | TBD | TBD | TBD |
| **Kernel Booting (Peak)** | TBD | TBD | TBD |
| **Kernel Idling** | TBD | TBD | TBD |

The current will fluctuate as various activates occur, such as the LEDs on and SD card accesses.

### 6.1.6 Processor Interfaces

The processor interacts with the **TPS65217C** via several different signals. Each of these signals is described below.

#### *6.1.6.1 I2C0*

I2C0 is the control interface between the processor and the **TPS65217C**. It allows the processor to control the registers inside the **TPS65217C** for such things as voltage scaling and switching of the input rails.

#### *6.1.6.2 PMC_POWR_EN*

ON power up the **VDD_RTC** rail activates first. After the RTC circuitry in the processor has activated it instructs the **TPS65217C** to initiate a full power up cycle by activating the **PMIC_POWR_EN** signal by taking it HI.

#### *6.1.6.3 LDO_GOOD*

This signal connects to the **RTC_PORZn** signal, RTC power on reset. As the RTC circuitry come sup first, this signal indicated that the LDOs, the 1.8V VRTC rail, is up and stable. This starts the power up process.

#### *6.1.6.4 PMIC_PGOOD*

Once all the rails are up, the PMIC_PGOOD signal goes high. This release the PORZn signal on the processor which was holding the processor reset.

#### *6.1.6.5 WAKEUP*

The WAKEUP signal from the **TPS65217C** is connected to the **EXT_WAKEUP** signal on the processor. This is used to wake up the processor when it I sin a sleep mode and an event from the **TPS65217C**, such as the power button, is pressed.

#### *6.1.6.6 PMIC_INT*

The **PMIC_INT** signal is an interrupt signal to the processor. If the power button feature is used, pressing he power button will send an interrupt to the processor allowing it to implement a power down mode in an orderly fashion.

### 6.1.7 Power Rails

The **Figure 11** shows the connections of each of the rails to the **TPS65217C**.

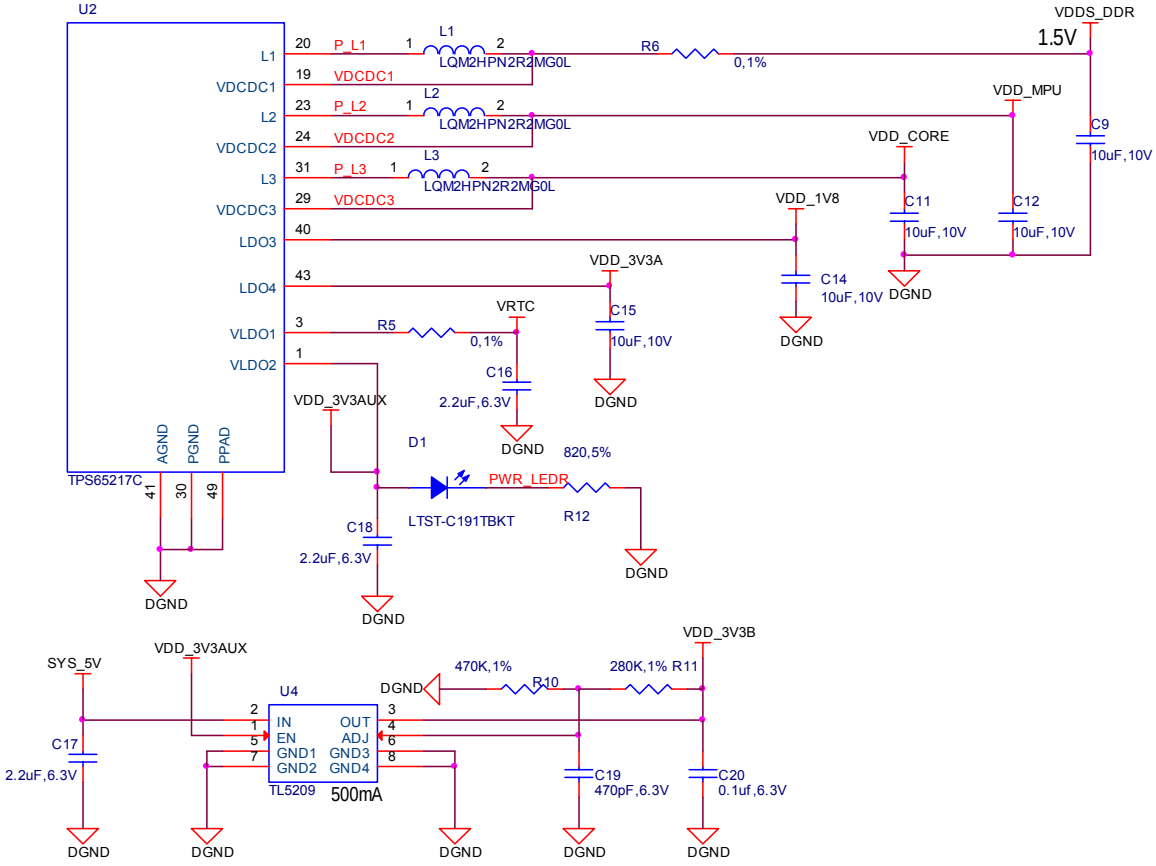


**Figure 11. Power Rails**

#### *6.1.7.1 VRTC Rail*

The **VRTC** rail is a 1.8V rail that is the first rail to come up in the power sequencing. It provides power to the RTC domain on the **XAM3359ZCZ** processor and the I/O rail of the **TPS65217C**. It can deliver up to 250mA maximum.

#### *6.1.7.2 VDD_3V3A Rail*

The **VDD_3V3A** rail is supplied by the **TPS65217C** and provides the 3.3V for the processor rails and can provide up to 400mA.

#### *6.1.7.3 VDD_3V3B Rail*

The current supplied by the **VDD_3V3A** rail is not sufficient to power all of the 3.3V rails on the board. So a second LDO is supplied, U4, a **TL5209A**, which sources the **VDD_3V3B** rail. It is powered up just after the **VDD_3V3A** rail.

#### *6.1.7.4 VDD_1V8 Rail*

The **VDD_1V8** rail can deliver up to 400mA and provides the power required for the 1.8V rails on the processor. This rail is not accessible for use anywhere else on the board.

#### *6.1.7.5 VDD_CORE Rail*

The **VDD_CORE** rail can deliver up to 1.2A at 1.1V. This rail is not accessible for use anywhere else on the board and only connects to the processor. This rail is fixed at 1.1V and is not scaled.

#### *6.1.7.6 VDD_MPU Rail*

The **VDD_MPU** rail can deliver up to 1.2A. This rail is not accessible for use anywhere else on the board and only connects to the processor. This rail defaults to 1.1V and can be scaled up to allow for higher frequency operation. Changing of the voltage is set via the I2C interface from the processor.

#### *6.1.7.7 VDDS_DDR Rail*

The **VDDS_DDR** rail defaults to **1.5V** to support the DDR3 rails and can deliver up to 1.2A. It is possible to adjust this voltage rail down to **1.35V** for lower power operation of the DDR3L device. Only DDR3L devices can support this voltage setting of 1.35V.

#### *6.1.7.8 Power Sequencing*

The power up process is made up of several stages and events. **Figure 12** is the events that make up the power up process for the system.

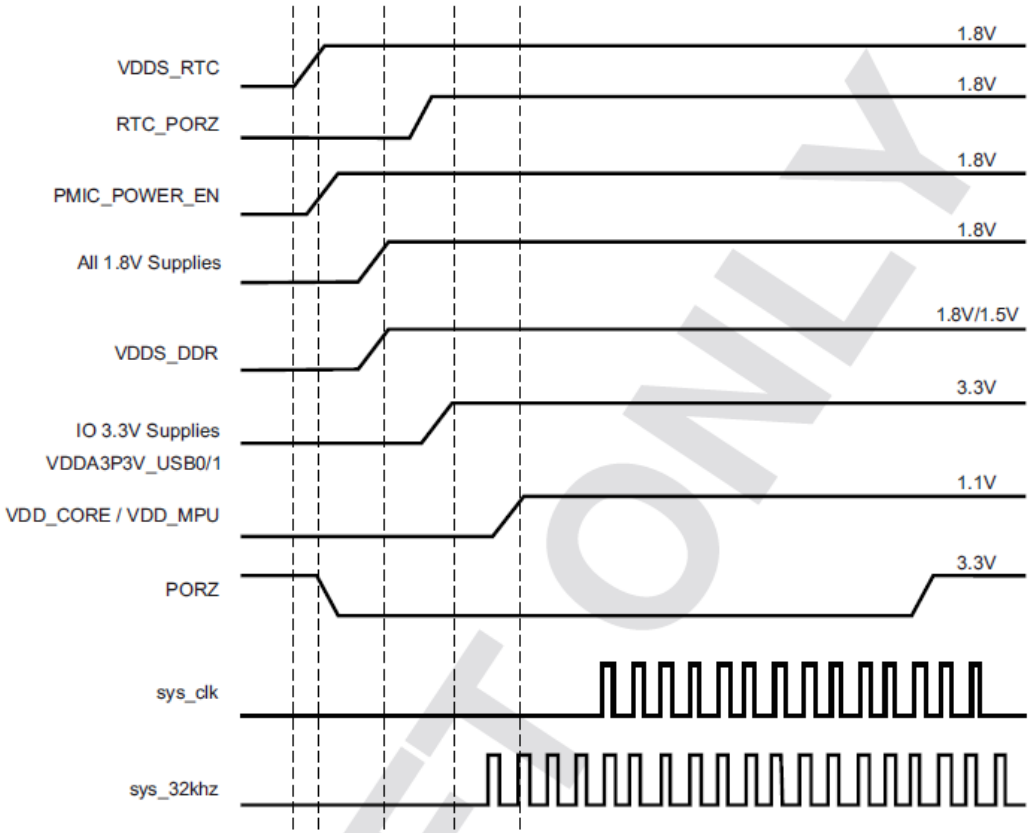


**Figure 12. Power Sequencing**

**Figure 13** is the way the **TPS65217C** powers up and the voltages on each rail. The power sequencing starts at 15 and then goes to one. That is the way the TPS65217C is configured.

| TPS65217C (Targeted at AM335x - ZCZ) | |
|---|---|
| **VOLTAGE (V)** | **SEQUENCE (STROBE)** |
| 1.5 | 1 |
| 1.1 | 5 |
| 1.1 | 5 |
| 1.8 | 15 |
| 3.3 | 3 |
| 1.8 (LDO, 400 mA) | 2 |
| 3.3 (LDO, 400 mA) | 4 |

**Figure 13. Power Sequencing**

### 6.1.8 Power LED

The power LED is a blue LED that will turn on once the **TPS65217C** has finished the power up procedure. If you ever see the LED flash once, that means that the **TPS65217C** started the process and encountered an issue that caused it to shut down. The connection of the LED is shown in **Figure 8**.

### 6.1.9 TPS65217C Power Up Process

**Figure 14** shows the interface between the **TPS65217C** and the processor.

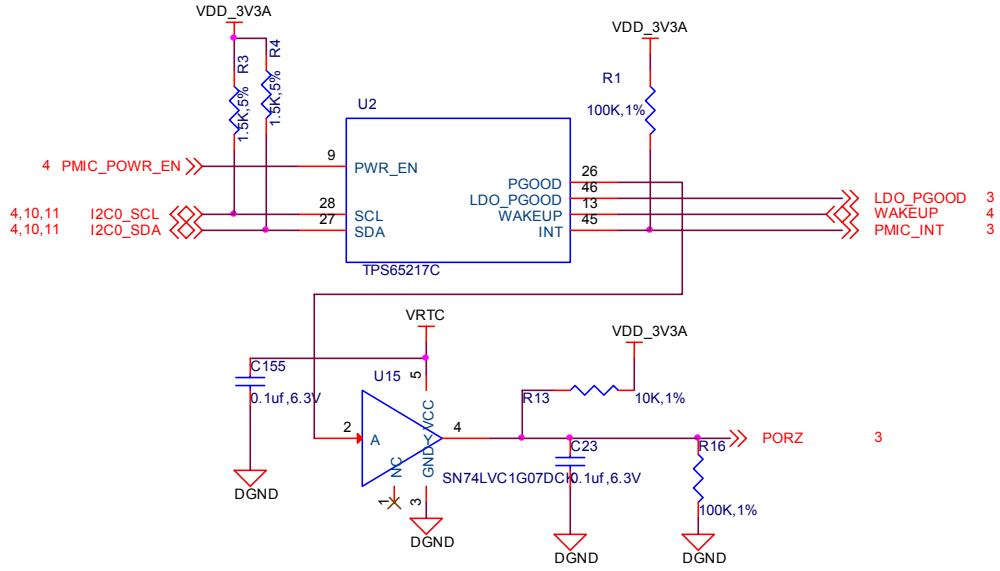


**Figure 14. Power Processor Interfaces**

When voltage is applied, DC or USB, the **TPS65217C** connects the power to the SYS output pin which drives the switchers and LDOS in the **TP65217C**.

At power up all switchers and LDOs are off except for the **VRTC LDO** (1.8V), which provides power to the VRTC rail and controls the **RTC_PORZ** input pin to the processor, which starts the power up process of the processor. Once the RTC rail powers up, the **RTC_PORZ** pin of the processor is released.

Once the **RTC_PORZ** reset is released, the processor starts the initialization process. After the RTC stabilizes, the processor launches the rest of the power up process by activating the **PMIC_PWR_EN** signal that is connected to the **TPS65217C** which starts the **TPS65217C** power up process.

The **LDO_PGOOD** signal is provided by the **TPS65217C** to the processor. As this signal is 1.8V from the **TPS65217C** by virtue of the TPS65217C VIO rail being set to

1.8V, and the **RTC_PORZ** signal on the processor is 3.3V, a voltage level shifter, **U4**, is used. Once the LDOs and switchers are up on the **TPS65217C**, this signal goes active releasing the processor. The LDOs on the **TPS65217C** are used to power the VRTC rail on the processor.

### 6.1.10 Processor Control Interface

**Figure 11** above shows two interfaces between the processor and the **TPS65217C** used for control after the power up sequence has completed.

The first is the **I2C0** bus. This allows the processor to turn on and off rails and to set the voltage levels of each regulator to supports such things as voltage scaling.

The second is the interrupt signal. This allows the T**PS65217C** to alert the processor when there is an event, such as when the optional power button is pressed. The interrupt is an open drain output which makes it easy to interface to 3.3V of the processor.

### 6.1.11 Low Power Mode Support

This section covers three general power down modes that are available. These modes are only described form a Hardware perspective as it relates to the HW design.

#### *6.1.11.1 RTC Only*

In this mode all rails are turned off except the **VDD_RTC**. The processor will need to turn off all the rails to enter this mode. The **VDD_RTC** staying on will keep the RTC active and provide for the wakeup interfaces to be active to respond to a wake up event.

#### *6.1.11.2 RTC Plus DDR*

In this mode all rails are turned off except the **VDD_RTC** and the **VDDS_DDR**, which powers the DDR3 memory. The processor will need to turn off all the rails to enter this mode. The **VDD_RTC** staying on will keep the RTC active and provide for the wakeup interfaces to be active to respond to a wake up event.

The **VDDS_DDR** rail to the DDR3 is provided by the 1.5V rail of the **TPDS65217C** and with VDDS_DDR active, the DDR3 can be placed in a self refresh mode by the processor prior to power down which allows the memory data to be saved.

#### *6.1.11.3 Voltage Scaling*

For a mode where the lowest power is possible without going to sleep, this mode allows the voltage on the ARM processor to be lowered along with slowing the processor

frequency down. The I2C0 bus is used to control the voltage scaling function in the **TPS65217C**.

## 6.2 XAM3359ZCZ Processor

The board is designed to use the AM3358 series processors in the 15 x 15 package. The initial units built will use the XAM3359AZC processor from TI. This is the same processor as used on the original BeagleBone except for a different revision. Later, we will switch to the AM338 device.

### 6.2.1 Description

**Figure 15** is a high level block diagram of the processor. For more information on the processor, go to http://www.ti.com/product/am3358.

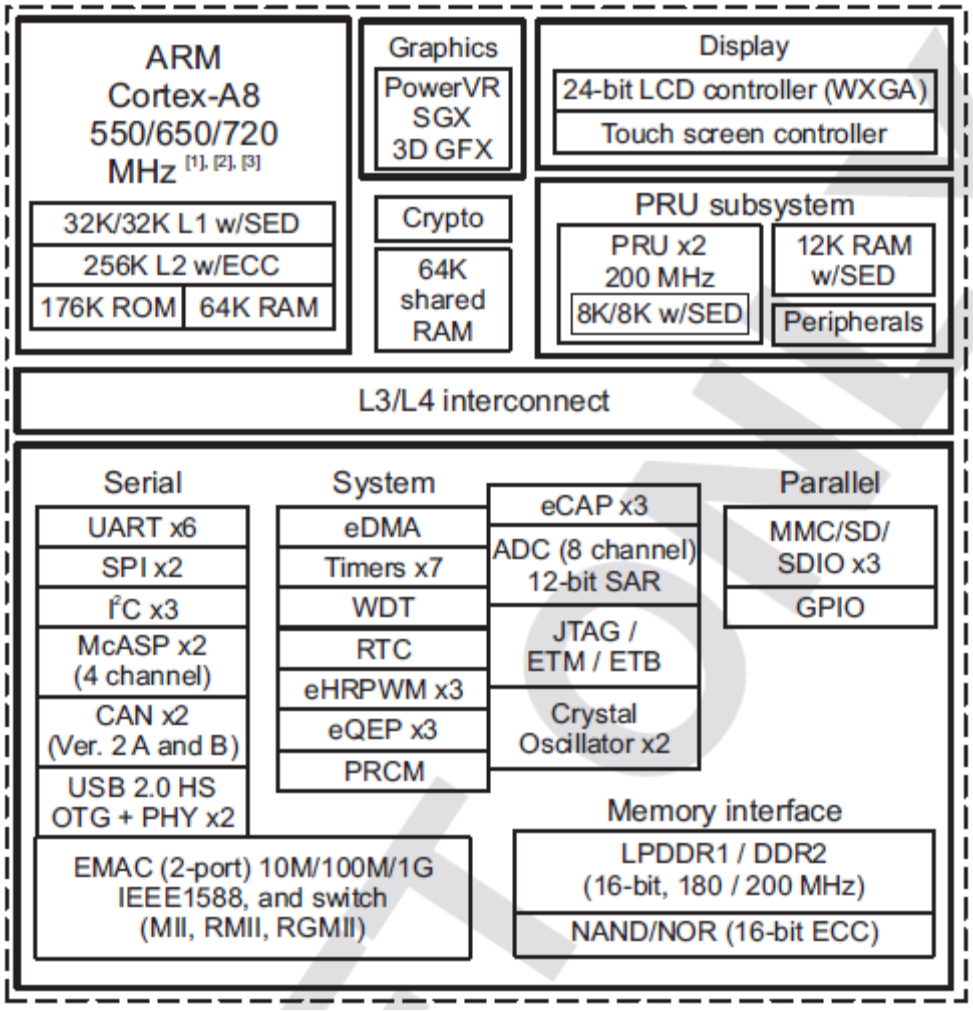


**Figure 15. XAM3359 Block Diagram**

### 6.2.2 High Level Features

**Table 4** below shows a few of the high level features of the AM3358 processor.

**Table 4. Processor Features**

| | | | |
|---|---|---|---|
| Operating Systems | Linux, Android, Windows Embedded CE | MMC/SD | 3 |
| Standby Power | 7 mW | CAN | 2 |
| ARM CPU | 1 ARM Cortex-A8 | UART (SCI) | 6 |
| ARM MHz (Max.) | 275,500,600,800 | ADC | 8-ch 12-bit |
| ARM MIPS (Max.) | 1000,1200,2000 | PWM (Ch) | 3 |
| Graphics Acceleration | 1 3D | eCAP | 3 |
| Other Hardware Acceleration | 2 PRU-ICSS,Crypto Accelerator | eQEP | 3 |
| On-Chip L1 Cache | 64 KB (ARM Cortex-A8) | RTC | 1 |
| On-Chip L2 Cache | 256 KB (ARM Cortex-A8) | I2C | 3 |
| Other On-Chip Memory | 128 KB | McASP | 2 |
| Display Options | LCD | SPI | 2 |
| General Purpose Memory | 1 16-bit (GPMC, NAND flash, NOR Flash, SRAM) | DMA (Ch) | 64-Ch EDMA |
| DRAM | 1 16-bit (LPDDR-400, DDR2-532, DDR3-606) | IO Supply (V) | 1.8V(ADC),3.3V |
| USB | 2 | Operating Temperature Range (C) | -40 to 90 |

### 6.2.3 Documentation

Full documentation for the XAM3359 processor can be found on the TI website at http://www.ti.com/product/am3359. Make sure that you always use the latest datasheets and Technical Reference Manuals (TRM).

## 6.3 DDR3 Memory

The BeagleBone Black uses a single MT41K256M16HA-125 512MB DDR3L device from Micron that interfaces to the processor over 16 data lines, 16 address lines, and 14 control lines. The following sections provide more details on the design.

### 6.3.1 Memory Device

The design will support standard DDR3 and DDR3L x16 devices. A single x16 device is used on the board and there is no support for two x8 devices. The DDR3 devices work at 1.5V and the DDR3 devices can work down to 1.35V to achieve lower power. The specific Micron device used is the MT41K256M16HA-125. It comes in a 96-BALL FBGA package with 0.8 mil pitch. Other standard DDR3 devices can also be supported, but the DDR3L is the lower power device and was chosen for its ability to work at 1.5V or 1.35V. The standard frequency that the DDR3 is run at is 303MHZ.

### 6.3.2 DDR3 Memory Design

**Figure 16** is the schematic for the DDR3L memory device. Each of the groups of signals is described in the following lines.

**Address Lines:** Provide the row address for ACTIVATE commands, and the column address and auto precharge bit (A10) for READ/WRITE commands, to select one location out of the memory array in the respective bank. A10 sampled during a PRECHARGE command determines whether the PRECHARGE applies to one bank (A10 LOW, bank selected by BA[2:0]) or all banks (A10 HIGH). The address inputs also provide the op-code during a LOAD MODE command. Address inputs are referenced to VREFCA. A12/BC#: When enabled in the mode register (MR), A12 is sampled during READ and WRITE commands to determine whether burst chop (on-the-fly) will be performed (HIGH = BL8 or no burst chop, LOW = BC4 burst chop).

**Bank Address Lines:** BA[2:0] define the bank to which an ACTIVATE, READ, WRITE, or PRECHARGE command is being applied. BA[2:0] define which mode register (MR0, MR1, MR2, or MR3) is loaded during the LOAD MODE command. BA[2:0] are referenced to $V_{REFCA}$.

**CK and CK# Lines:** are differential clock inputs. All address and control input signals are sampled on the crossing of the positive edge of CK and the negative edge of CK#. Output data strobe (DQS, DQS#) is referenced to the crossings of CK and CK#.

**Clock Enable Line:** CKE enables (registered HIGH) and disables (registered LOW) internal circuitry and clocks on the DRAM. The specific circuitry that is enabled/disabled is dependent upon the DDR3 SDRAM configuration and operating mode. Taking CKE LOW provides PRECHARGE power-down and SELF REFRESH operations (all banks idle) or active power-down (row active in any bank). CKE is synchronous for power-down entry and exit and for self refresh entry. CKE is asynchronous for self refresh exit.

Input buffers (excluding CK, CK#, CKE, RESET#, and ODT) are disabled during power-down. Input buffers (excluding CKE and RESET#) are disabled during SELF REFRESH. CKE is referenced to $V_{REFCA}$.

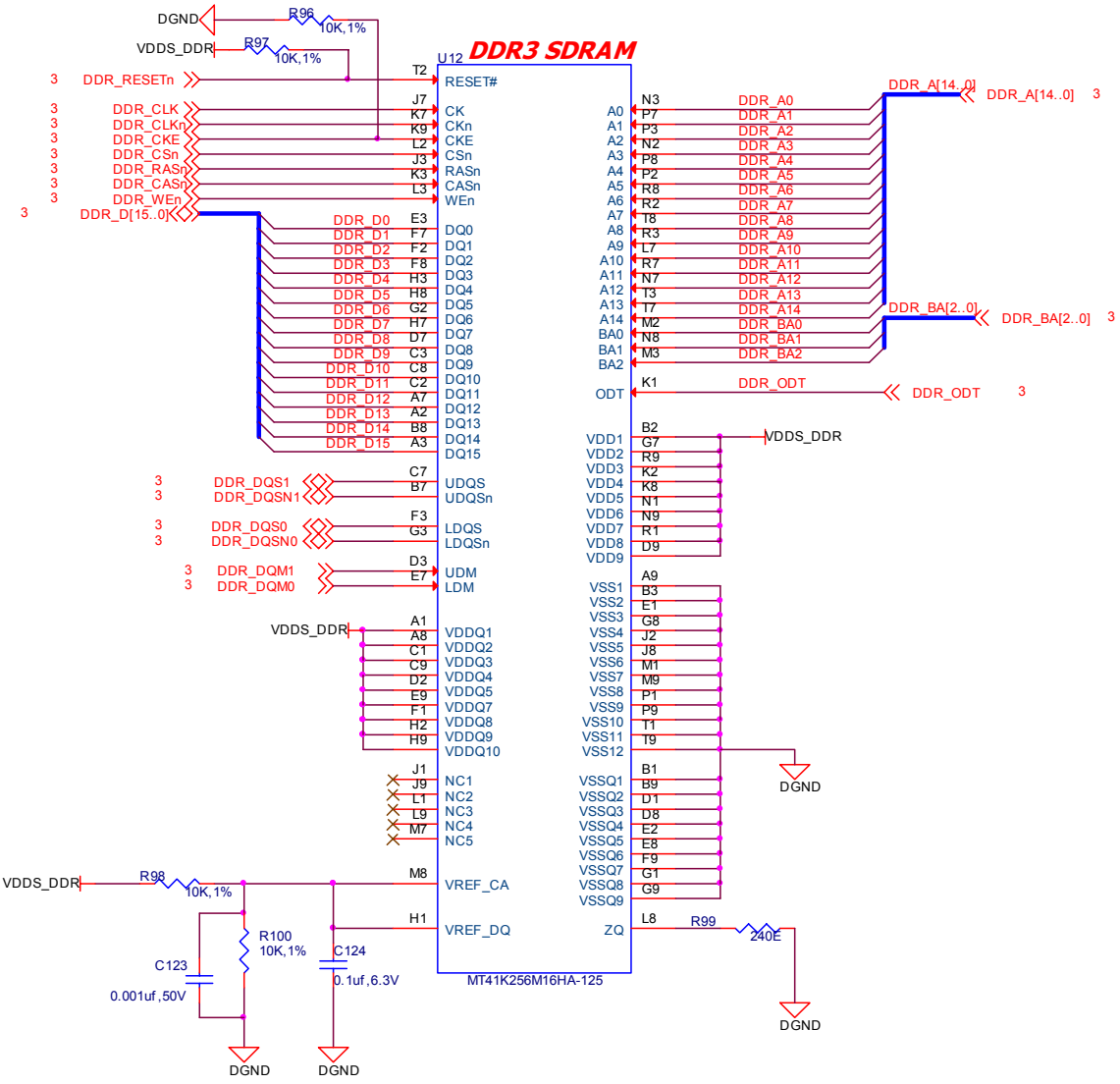


**Figure 16. DDR3 Memory Design**

**Chip Select Line:** CS# enables (registered LOW) and disables (registered HIGH) the command decoder. All commands are masked when CS# is registered HIGH. CS# provides for external rank selection on systems with multiple ranks. CS# is considered part of the command code. CS# is referenced to $V_{REFCA}$.

**Input Data Mask Line:** DM is an input mask signal for write data. Input data is masked when DM is sampled HIGH along with the input data during a write access. Although the DM ball is input-only, the DM loading is designed to match that of the DQ and DQS balls. DM is referenced to VREFDQ.

**On-die Termination Line:** ODT enables (registered HIGH) and disables (registered LOW) termination resistance internal to the DDR3 SDRAM. When enabled in normal operation, ODT is only applied to each of the following balls: DQ[7:0], DQS, DQS#, and DM for the x8; DQ[3:0], DQS, DQS#, and DM for the x4. The ODT input is ignored if disabled via the LOAD MODE command. ODT is referenced to VREFCA.

### 6.3.3 Power Rails

The **DDR3L** memory device and the DDR3 rails on the processor are supplied by the **TPS65217C**. Default voltage is 1.5V but can be scaled down to 1.35V if desired.

### 6.3.4 VREF

The **VREF** signal is generated from a voltage divider on the **VDDS_DDR** rail that powers the processor DDR rail and the DDR3L device itself. **Figure 17** below shows the configuration of this signal and the connection to the DDDR3 memory device and the processor.

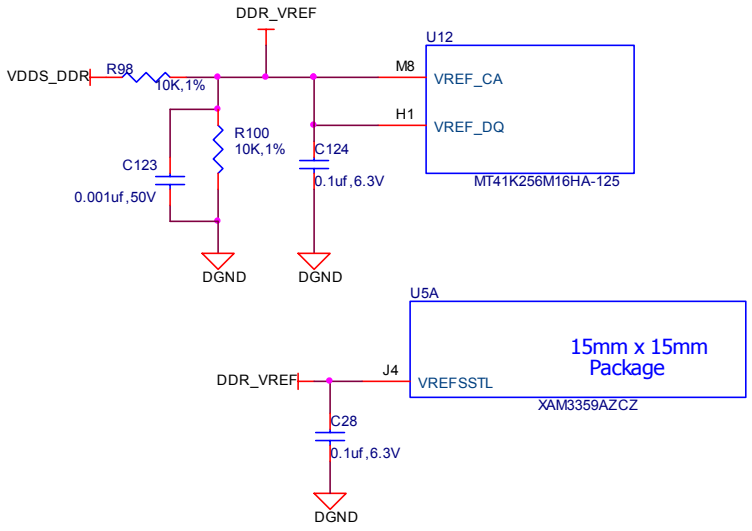


**Figure 17. DDR3 VREF Design**

## 6.4 eMMC Memory

The eMMC is a communication and mass data storage device that includes a Multi-MediaCard (MMC) interface, a NAND Flash component, and a controller on an advanced 11-signal bus, which is compliant with the MMC system specification. The nonvolatile eMMC draws no power to maintain stored data, delivers high performance across a wide range of operating temperatures, and resists shock and vibration disruption.

One of the issues faced with SD cards is that across the different brands and even within the same brand, performance can vary. Cards use different controllers and different memories, all of which can have bad locations that the controller handles, but the controller may be optimized for reads or writes. You never know what you will be getting. This can lead to varying rates of performance. The eMMC card is a known controller and when coupled with the 8bit mode, 8 bits of data instead of 4, you get double the performance which should result in quicker boot times.

The following sections describe the design and device that is used on the BeagleBone Black to implement this interface.

### 6.4.1 eMMC Device

The device used in a Micron **MTFC2GMTEA-0F_WT** 2GB eMMC device. This is a new device and so for documentation and support, you will need to contact your local Micron representative.

The package is a 153 ball WFBGA device. The footprint on the BeagleBone Black for this device supports 4GB and 8GB devices. As this is a JEDEC standard, there are other suppliers that may work in this design as well. The only device that has been tested is the **MTFC2GMTEA-0F_WT**.

### 6.4.2 eMMC Circuit Design

**Figure 18** is the design of the eMMC circuitry. The eMMC device is connected to the MMC1 port on the processor. MMC0 is still used for the uSD card as is currently done on the original BeagleBone.

The device runs at 3.3V both internally and the external I/O rails. The VCCI is an internal voltage rail to the device. The manufacturer recommends that a 1uf capacitor be attached to this rail, but a 2.2uF was chosen to provide a little margin.

Pullup resistors are used to increase the rise time on the signals to compensate for any capacitance on the board.

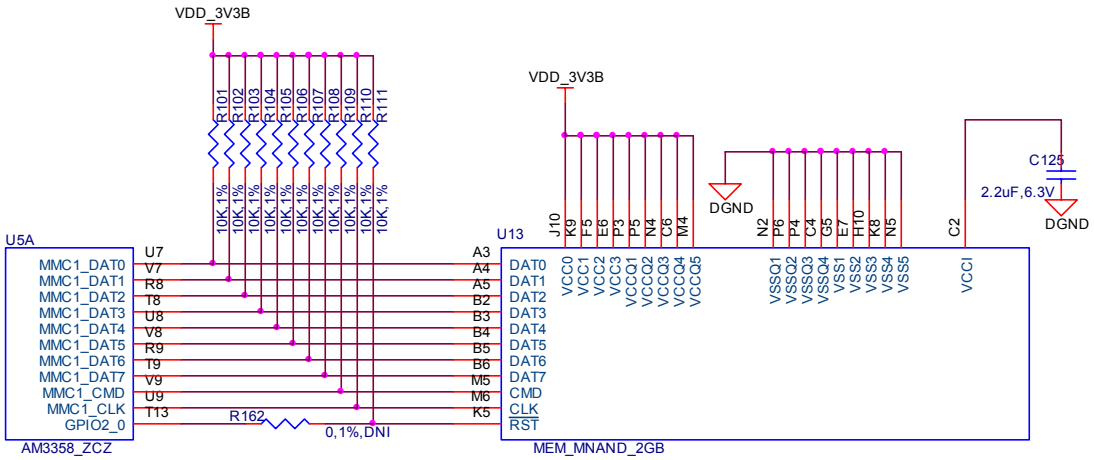


**Figure 18. eMMC Memory Design**

The pins used by the eMMC1 in the boot mode are listed below in **Table 5**.

**Table 5. eMMC Boot Pins**

| Signal name | Pin Used in Device |
|---|---|
| clk | gpmc_csn1 |
| cmd | gpmc_csn2 |
| dat0 | gpmc_ad0 |
| dat1 | gpmc_ad1 |
| dat2 | gpmc_ad2 |
| dat3 | gpmc_ad3 |

For eMMC devices the ROM will only support raw mode. The ROM Code reads out raw sectors from image or the booting file within the file system and boots from it. In raw mode the booting image can be located at one of the four consecutive locations in the main area: offset 0x0 / 0x20000 (128 KB) / 0x40000 (256 KB) / 0x60000 (384 KB). For this reason, a booting image shall not exceed 128KB in size. However it is possible to flash a device with an image greater than 128KB starting at one of the aforementioned locations. Therefore the ROM Code does not check the image size. The only drawback is that the image will cross the subsequent image boundary. The raw mode is detected by reading sectors #0, #256, #512, #768. The content of these sectors is then verified for presence of a TOC structure. In the case of a **GP Device**, a Configuration Header (CH) **must** be located in the first sector followed by a **GP header**. The CH might be void (only containing a CHSETTINGS item for which the Valid field is zero).

The ROM only supports the 4-bit mode. After the initial boot, the switch can be made to 8-bit mode for increasing the overall performance of the eMMC interface.

## 6.5 Micro Secure Digital

The uSD connector on the board will support a uSD card that can be used for booting or file storage on the BeagleBone Black.

### 6.5.1 uSD Design

**Figure 19** below is the design of the uSD interface.

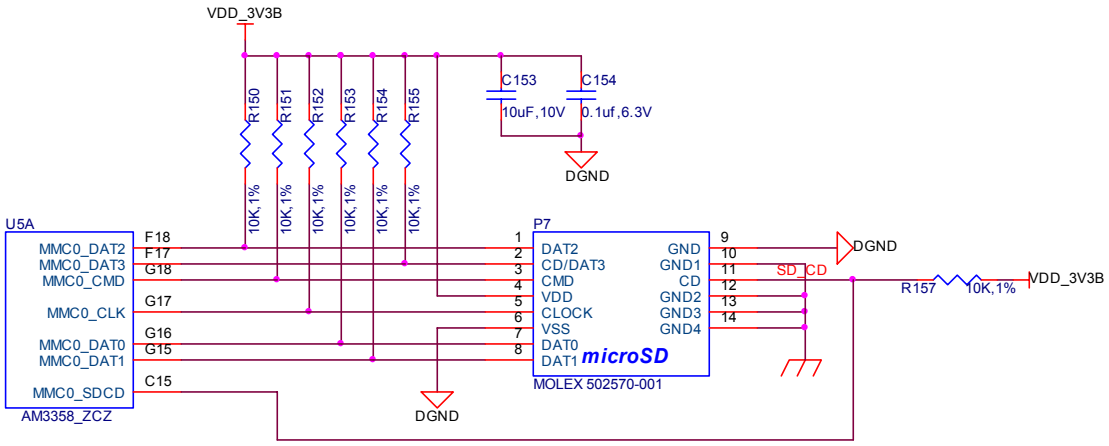


**Figure 19. uSD Design**

The signals **MMC0-3** are the data lines for the transfer of data between the processor and the uSD connector.

The **MMC0_CLK** signal clocks the data in and out of the uSD card.

The **MMCO_CMD** signal indicates that a command versus data is being sent.

There is no separate card detect pin in the uSD specification. It uses **MMCO_DAT3** for that function. However, most uSD connectors still supply a CD function on the connectors. In the BeagleBone Black design, this pin is connected to the **MMC0_SDCD** pin for use by the processor. You can also change the pin to **GPIO0_6**, which is able to wake up the processor from a sleep mode when an SD card is inserted into the connector.

Pullup resistors are provided on the signals to increase the rise times of the signals to overcome PCB capacitance.

Power is provided from the **VDD_3V3B** rail and a 10uf capacitor is provided for filtering.

## 6.6 User LEDs

There are four user LEDs on the BeagleBone Black. These are connected to GPIO pins on the processor. **Figure 20** shows the interfaces for the user LEDs.

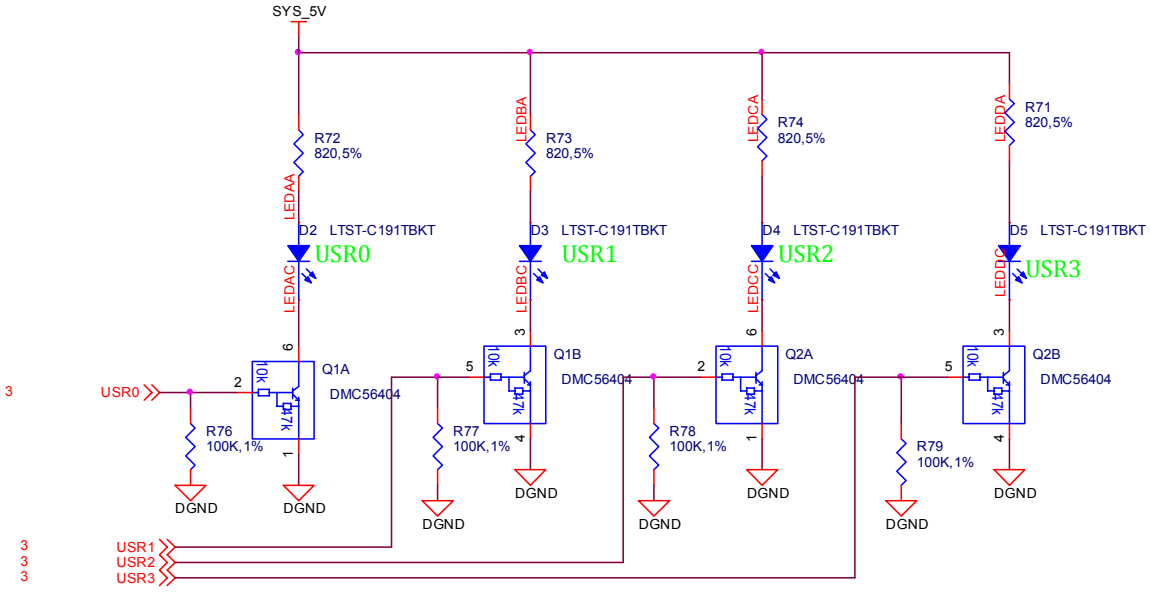


**Figure 20. User LEDs**

## 6.7 Boot Configuration

The design supports two groups of boot options on the board. The user can switch between these modes via the Boot button. The primary boot source is the onboard eMMc device. By holding the Boot button, the user can force the board to boot from the uSD slot. This enables the eMMC to be overwritten when needed or to just boot an alternate image. The following sections describe how the boot configuration works.

### 6.7.1 Boot Configuration Design

**Figure 21** shows the circuitry that is involved in the boot configuration process. On power up, these pins are read by the processor to determine the boot order. S2 is used to change the level of one bit from Hi to LO which changes the boot order.

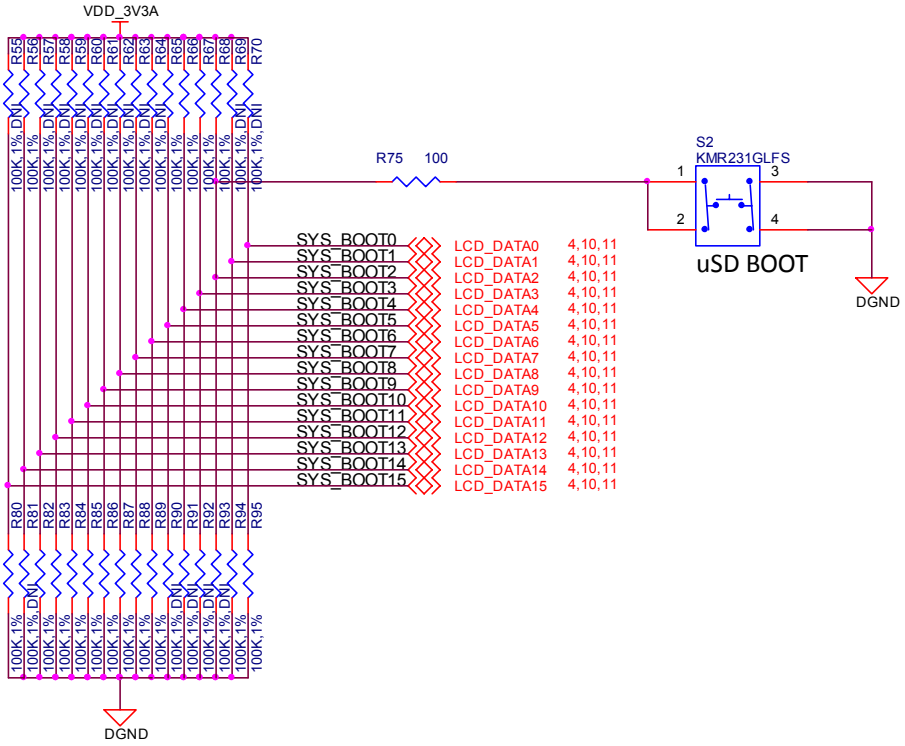


**Figure 21. Processor Boot Configuration Design**

It is possible to override these setting via the expansion headers. But be careful not to add too much load such that it could interfere with the operation of the HDMI interface or LCD panels.

### 6.7.2 Boot Options

Based on the selected option found in **Figure 19** below, each of the boot sequences for each of the two settings is shown.

| SYSBOOT[15:14] | SYSBOOT[13:12] | SYSBOOT[11:10] | SYSBOOT[9] | SYSBOOT[8] | SYSBOOT[7:6] | SYSBOOT[5] | SYSBOOT[4:0] | Boot Sequence | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 00b = 19.2MHz<br>01b = 24MHz<br>10b = 25MHz<br>11b = 26MHz | 00b<br>(all other values reserved) | Don't care for ROM code | Don't care for ROM code | Don't care for ROM code | Don't care for ROM code | 0 = CLKOUT1 disabled<br>1 = CLKOUT1 enabled | 11100b | MMC1 | MMC0 | UART0 | USB0[5] |
| 00b = 19.2MHz<br>01b = 24MHz<br>10b = 25MHz<br>11b = 26MHz | 00b<br>(all other values reserved) | Don't care for ROM code | Don't care for ROM code | Don't care for ROM code | Don't care for ROM code | 0 = CLKOUT1 disabled<br>1 = CLKOUT1 enabled | 11000b | SPI0 | MMC0 | USB0[5] | UART0 |

**Figure 22. Processor Boot Configuration**

The first row in **Figure 22** is the default setting. On boot, the processor will look for the eMMC on the MMC1 port first, followed by USB0 and UART0. In the event there is no uSD card and the eMMC is empty, UART0 or USB0 could be used as the board source.

If you have a uSD card from which you need to boot from, hold the boot button down. On boot, the processor will look for the SPIO0 port first, then uSD on the MMC0 port, followed by USB0 and UART0. In the event there is no uSD card and the eMMC is empty, USB0 or UART0 could be used as the board source.

## 6.8 10/100 Ethernet

The BeagleBone Black is equipped with a 10/100 Ethernet interface. It uses the same PHY as is used on the original BeagleBone. The design is described in the following sections.

### 6.8.1 Ethernet Processor Interface

**Figure 23** shows the connections between the processor and the PHY. The interface is in the MII mode of operation.

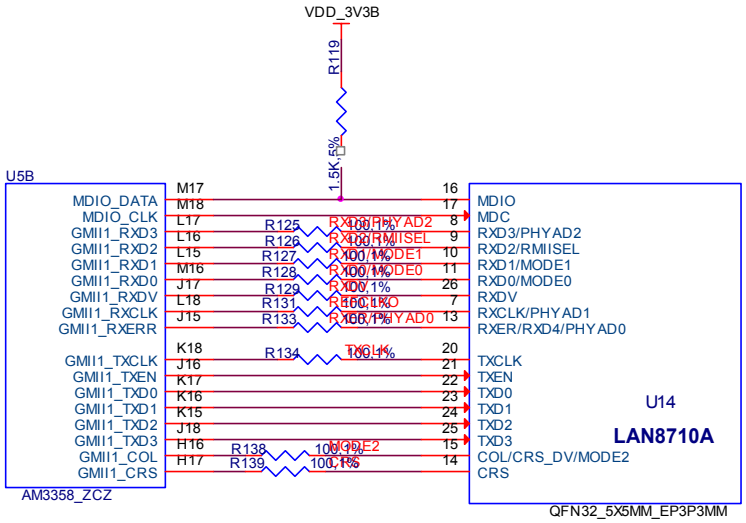


**Figure 23. Ethernet Processor Interface**

### 6.8.2 Ethernet Connector Interface

The off board side of the PHY connections are shown in **Figure 24** below.

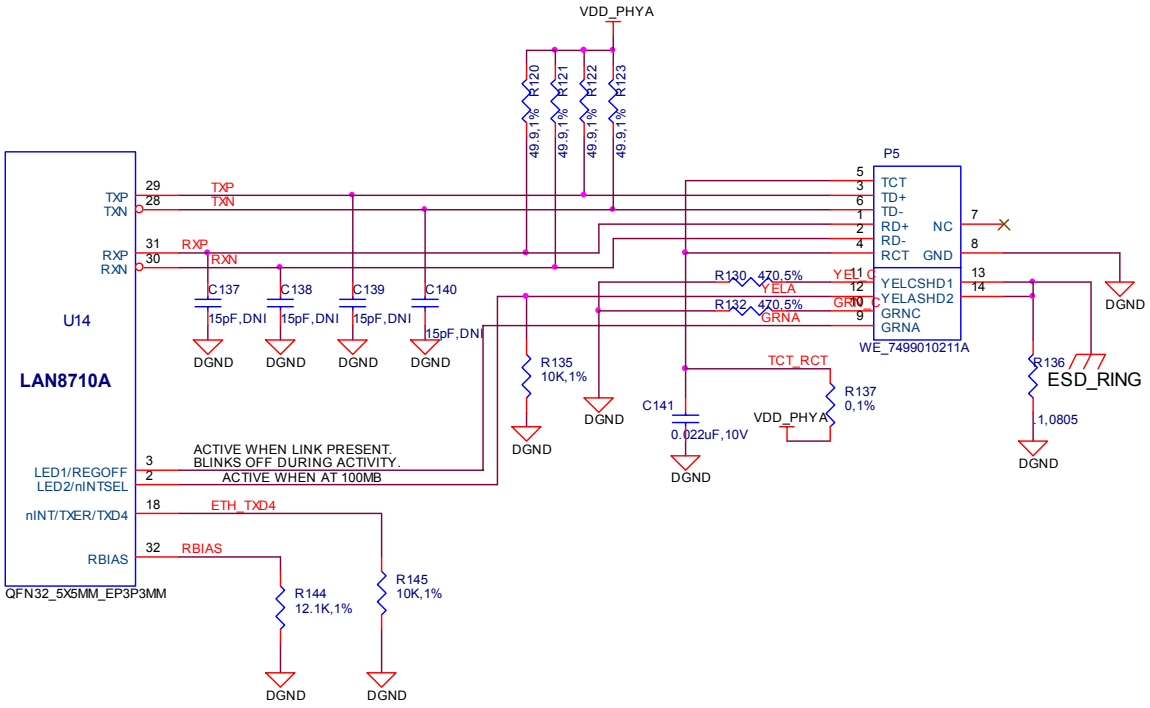


**Figure 24. Ethernet Connector Interface**

### 6.8.3 Ethernet PHY Power, Reset, and Clocks

**Figure 25** show the power, reset, and lock connections to the **LAN8710A** PHY. Each of these areas is discussed in more detail in the following sections.

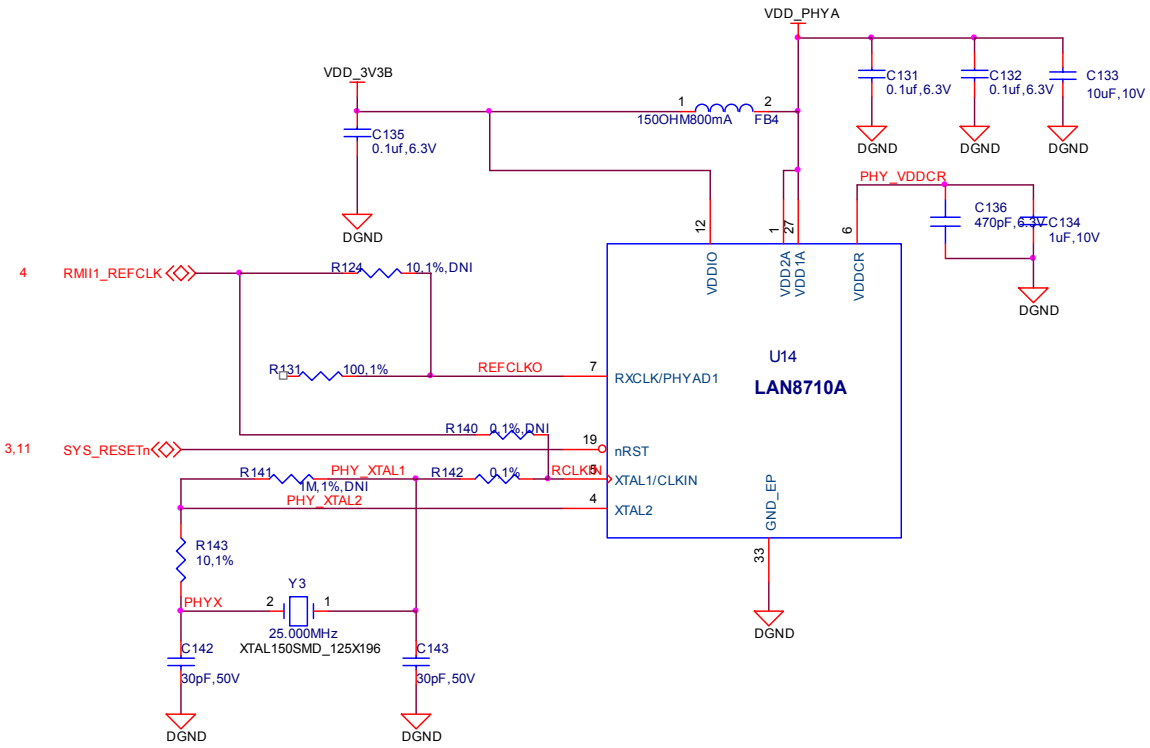


**Figure 25. Ethernet PHY, Power, Reset, and Clocks**

#### *6.8.3.1 VDD_3V3B Rail*

The VDD_3V3B rail is the main power rail for the **LAN8710A**. It originates at the VD_3V3B regulator and is the primary rail that supports all of the peripherals on the board. This rail also supplies the VDDIO rails which set the voltage levels for all of the I/O signals between the processor and the **LAN8710A**.

#### *6.8.3.2 VDD_PHYA Rail*

A filtered version of VDD_3V3B rail is connected to the VDD rails of the LAN8710 and the termination resistors on the Ethernet signals. It is labeled as **VDD_PHYA**. The filtering inductor helps block transients that may be seen on the VDD_3V3B rail.

#### 6.8.3.3 PHY_VDDCR Rail

The **PHY_VDDCR** rail originates inside the LAN8710A. Filter and bypass capacitors are used to filter the rail. Only circuitry inside the LAN8710A uses this rail.

#### 6.8.3.4 SYS_RESET

The reset of the LAN8710A is controlled via the **SYS_RESETn** signal, the main board reset line.

#### 6.8.3.5 Clock Signals

A crystal is used to create the clock for the LAN8710A. The processor uses the **RMII_RXCLK** signal to provide the clocking for the data between the processor and the LAN8710A.

### 6.8.4 LAN8710A Mode Pins

There are mode pins on the LAN8710A that sets the operational mode for the PHY when coming out of reset. These signals are also used to communicate between the processor and the LAN8710A. As a result, these signals can be driven by the processor which can cause the PHY not to be initiated correctly. To insure that this does not happen, three low value pull up resistors are used. **Figure 26** below shows the three mode pin resistors.

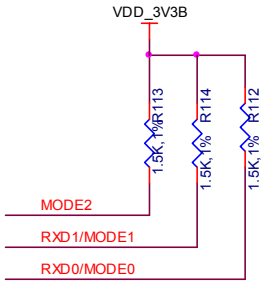


**Figure 26. Ethernet PHY Mode Pins**

This will set the mode to be 111, which enables all modes and enables auto-negotiation.

## 6.9 HDMI Interface

The BeagleBone Black has an onboard HDMI framer that converts the LCD signals and audio signals to drive a HDMI monitor. The design uses an NXP **TDA199988** HDMI Framer.

The following sections provide more detail into the design of this interface.

### 6.9.1 HDMI Framer

The **TDA19988** is a High-Definition Multimedia Interface (HDMI) 1.4a transmitter. It is backward compatible DVI 1.0 and can be connected to any DVI 1.0 or HDMI sink. The HDCP mode is not used in the design. The non-HDCP version of the device is used in the BeagleBone Black design.

This device provides additional embedded features like CEC (Consumer Electronic Control). CEC is a single bidirectional bus that transmits CEC over the home appliance network connected through this bus. This eliminates the need of any additional device to handle this feature. While this feature is supported in this device, as of this point, the SW to support this feature has not been implemented and Is not a feature that is considered critical.

It can be switched to very low power Standby or Sleep modes to save power when HDMI is not used. TDA19988 embeds I2C-bus master interface for DDC-bus communication to read EDID. This device can be controlled or configured via I2C-bus interface.

### 6.9.2 HDMI Video Processor Interface

The **Figure 27** shows the connections between the processor and the HDMI framer device. There are 16 bits of display data, 5-6-5 that is used to drive the framer. The reason for 16 bits I stat allows for compatibility with display and LCD capes already available on the original BeagleBone. The unused bits on the TDA19988 are tied low. In addition to the data signals are the VSYNC, HSYNC, DE, and PCLK signals that round out the video interface from the processor.

### 6.9.3 HDMI Control Processor Interface

In order to use the TDA19988, the processor needs to setup the device. This is done via the I2C interface between the processor and the TDA19988. There are two signals on the TDA19988 that could be used to set the address of the TDA19988. In this design they are both tied low. The I2C interface supports both 400kHz and 100KhZ operation. **Table 6** shows the I2C address.

**Table 6. TDA19988 I2C Address**

| HDMI core address | | | | | | | |
|---|---|---|---|---|---|---|---|
| A6 | A5 | A4 | A3 | A2 | A1 | A0 | R/W |
| 1 | 1 | 1 | 0 | 0 | X[1] | X[1] | 0/1 |

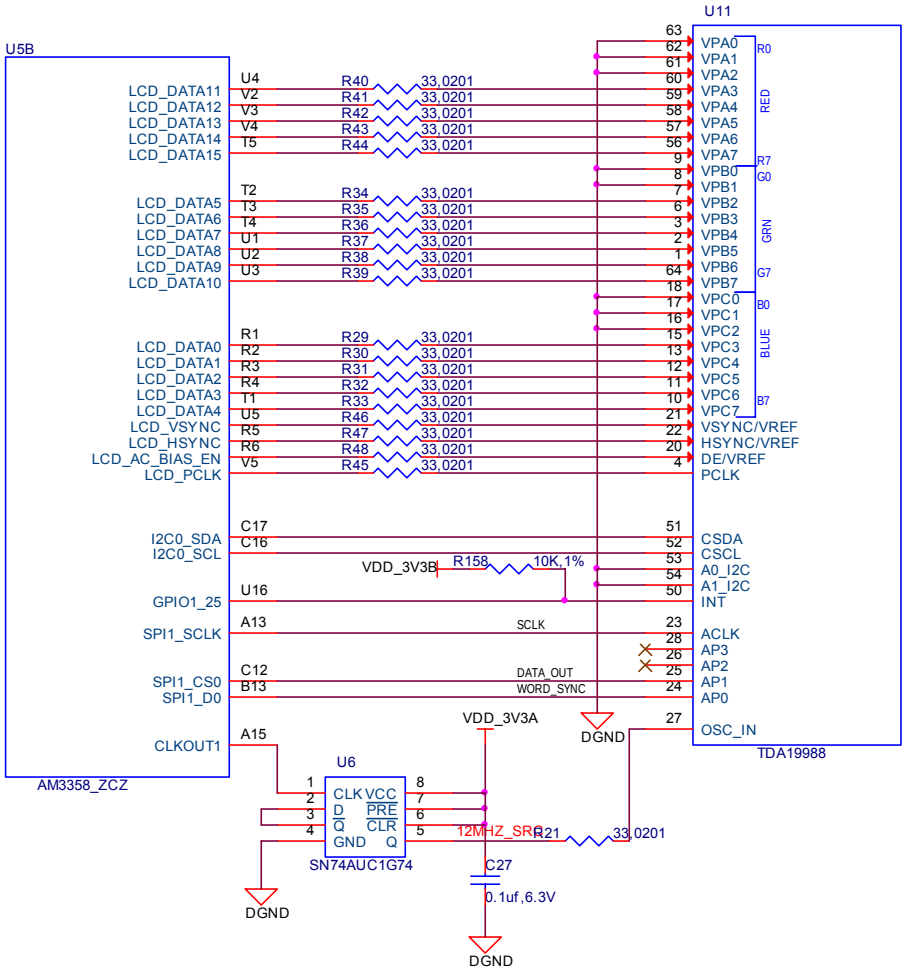


**Figure 27. HDMI Framer Processor Interface**

### 6.9.4 Interrupt Signal

There is a HDMI_INT signal that connects from the TDA19988 to the processor. This signal can be used to alert the processor in a state change on the HDMI interface.

### 6.9.5 Audio Interface

There is an I2S audio interface between the processor and the TDA19988. Stereo audio can be transported over the HDMI interface to an audio equipped display. In order to create the required clock frequencies, and external 24.576MHz oscillator, **Y4**, is used. From this clock, the processor generates the required clock frequencies for the TDA19988.

There are three signals used to pass data from the processor to the TDA199888. SCLK is the serial clock. SPI1_CS0 is the data pin to the TDA19888. SPI1_D0 is the word sync pin. These signals are configured as I2S interfaces.

### 6.9.6 Power Connections

**Figure 28** shows the power connections to the **TDA19988** device. All voltage rails for the device are at 1.8V. A filter is provided to minimize any noise from the 1.8V rail getting back into the device.

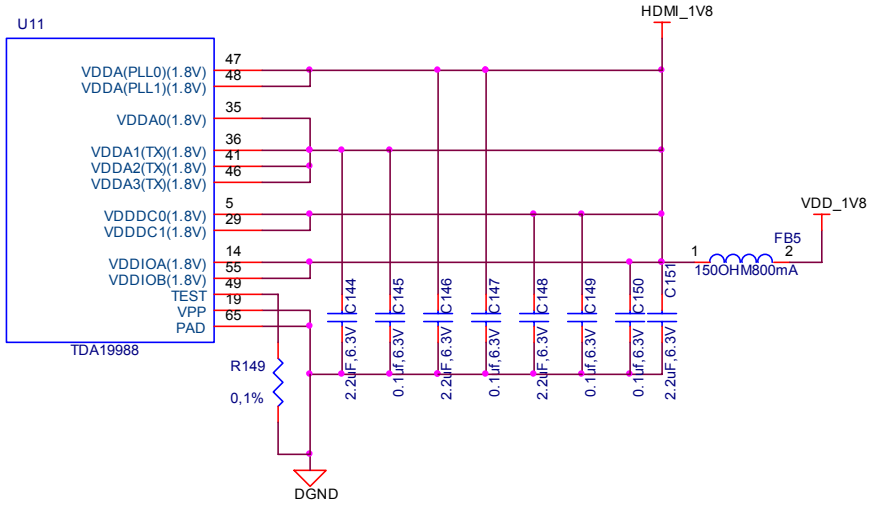


**Figure 28. HDMI Power Connections**

All of the interfaces between the processor and the TDA19988 are 3.3V tolerant allowing for direct connection.

### 6.9.7 HDMI Connector Interface

**Figure 29** shows the design of the interface between the HDMI Framer and the connector.

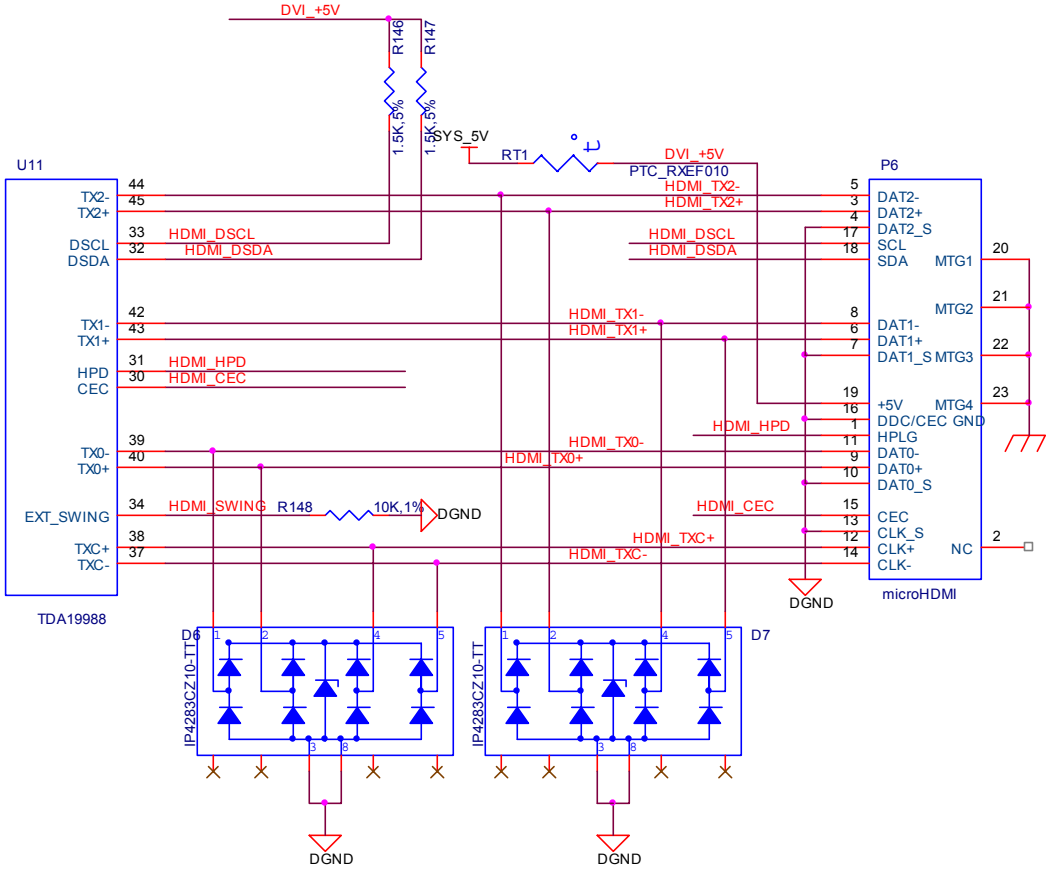


**Figure 29. Connector Interface Circuitry**

The connector for the HDMI interface is a microHDMI. It should be noted that this connector has a different pinout that the regular or mini HDMI connectors. D6 and D7 are ESD protection devices.

# 7.0 Connectors

This section describes each of the connectors on the board.

## 7.1 Expansion Connectors

The expansion interface on the board is comprised of two 46 pin connectors. All signals on the expansion headers are **3.3V** unless otherwise indicated.

**NOTE: Do not connect 5V logic level signals to these pins or the board will be damaged.**

**Figure 30** shows the location of the Expansion connectors.

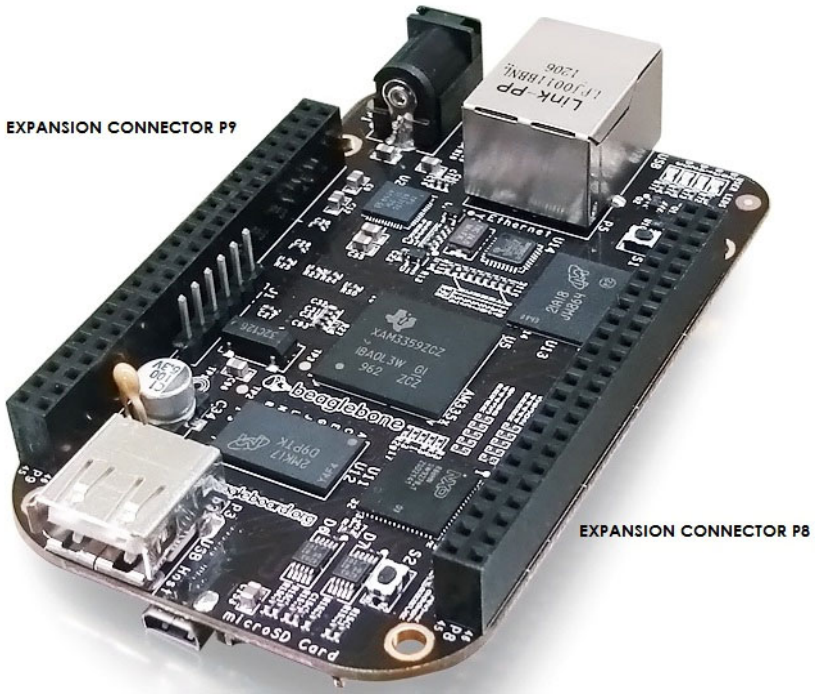


**Figure 30. Expansion Connector Location**

The location and spacing of the expansion headers are the same as on the original BeagleBone.

### 7.1.1 Connector P8

**Table 7** shows the pinout of the **P8** expansion header. Other signals can be connected to this connector based on setting the pin mux on the processor, but this is the default settings on power up. The SW is responsible for setting the default function of each pin. There are some signals that have not been listed here. Refer to the processor documentation for more information on these pins and detailed descriptions of all of the pins listed. In some cases there may not be enough signals to complete a group of signals that may be required to implement a total interface.

The **PROC** column is the pin number on the processor.
The **PIN** column is the pin number on the expansion header.
The **MODE** columns are the mode setting for each pin. Setting each mode to align with the mode column will give that function on that pin.

## Table 7. Expansion Header P8 Pinout

| PIN | PROC | NAME | MODE0 | MODE1 | MODE2 | MODE3 | MODE4 | MODE5 | MODE6 | MODE7 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1,2 | GND | | | | | | | | | |
| 3 | R9 | GPIO1_6 | gpmc_ad6 | mmc1_dat6 | | | | | | gpio1[6] |
| 4 | T9 | GPIO1_7 | gpmc_ad7 | mmc1_dat7 | | | | | | gpio1[7] |
| 5 | R8 | GPIO1_2 | gpmc_ad2 | mmc1_dat2 | | | | | | gpio1[2] |
| 6 | T8 | GPIO1_3 | gpmc_ad3 | mmc1_dat3 | | | | | | gpio1[3] |
| 7 | R7 | TIMER4 | gpmc_advn_ale | | timer4 | | | | | gpio2[2] |
| 8 | T7 | TIMER7 | gpmc_oen_ren | | timer7 | | | | | gpio2[3] |
| 9 | T6 | TIMER5 | gpmc_be0n_cle | | timer5 | | | | | gpio2[5] |
| 10 | U6 | TIMER6 | gpmc_wen | | timer6 | | | | | gpio2[4] |
| 11 | R12 | GPIO1_13 | gpmc_ad13 | lcd_data18 | mmc1_dat5 | mmc2_dat1 | eQEP2B_in | | | gpio1[13] |
| 12 | T12 | GPIO1_12 | GPMC_AD12 | LCD_DATA19 | MMC1_DAT4 | MMC2_DAT0 | EQEP2A_IN | | | gpio1[12] |
| 13 | T10 | EHRPWM2B | gpmc_ad9 | lcd_data22 | mmc1_dat1 | mmc2_dat5 | ehrpwm2B | | | gpio0[23] |
| 14 | T11 | GPIO0_26 | gpmc_ad10 | lcd_data21 | mmc1_dat2 | mmc2_dat6 | ehrpwm2_tripzone_in | | | gpio0[26] |
| 15 | U13 | GPIO1_15 | gpmc_ad15 | lcd_data16 | mmc1_dat7 | mmc2_dat3 | eQEP2_strobe | | | gpio1[15] |
| 16 | V13 | GPIO1_14 | gpmc_ad14 | lcd_data17 | mmc1_dat6 | mmc2_dat2 | eQEP2_index | | | gpio1[14] |
| 17 | U12 | GPIO0_27 | gpmc_ad11 | lcd_data20 | mmc1_dat3 | mmc2_dat7 | ehrpwm0_synco | | | gpio0[27] |
| 18 | V12 | GPIO2_1 | gpmc_clk_mux0 | lcd_memory_clk | gpmc_wait1 | mmc2_clk | | | mcasp0_fsr | gpio2[1] |
| 19 | U10 | EHRPWM2A | gpmc_ad8 | lcd_data23 | mmc1_dat0 | mmc2_dat4 | ehrpwm2A | | | gpio0[22] |
| 20 | V9 | GPIO1_31 | gpmc_csn2 | gpmc_be1n | mmc1_cmd | | | | | gpio1[31] |
| 21 | U9 | GPIO1_30 | gpmc_csn1 | gpmc_clk | mmc1_clk | | | | | gpio1[30] |
| 22 | V8 | GPIO1_5 | gpmc_ad5 | mmc1_dat5 | | | | | | gpio1[5] |
| 23 | U8 | GPIO1_4 | gpmc_ad4 | mmc1_dat4 | | | | | | gpio1[4] |
| 24 | V7 | GPIO1_1 | gpmc_ad1 | mmc1_dat1 | | | | | | gpio1[1] |
| 25 | U7 | GPIO1_0 | gpmc_ad0 | mmc1_dat0 | | | | | | gpio1[0] |
| 26 | V6 | GPIO1_29 | gpmc_csn0 | | | | | | | gpio1[29] |
| 27 | U5 | GPIO2_22 | lcd_vsync | gpmc_a8 | | | | | | gpio2[22] |
| 28 | V5 | GPIO2_24 | lcd_pclk | gpmc_a10 | | | | | | gpio2[24] |
| 29 | R5 | GPIO2_23 | lcd_hsync | gpmc_a9 | | | | | | gpio2[23] |
| 30 | R6 | GPIO2_25 | lcd_ac_bias_en | gpmc_a11 | | | | | | gpio2[25] |
| 31 | V4 | UART5_CTSN | lcd_data14 | gpmc_a18 | eQEP1_index | mcasp0_axr1 | uart5_rxd | | uart5_ctsn | gpio0[10] |
| 32 | T5 | UART5_RTSN | lcd_data15 | gpmc_a19 | eQEP1_strobe | mcasp0_ahclkx | mcasp0_axr3 | | uart5_rtsn | gpio0[11] |
| 33 | V3 | UART4_RTSN | lcd_data13 | gpmc_a17 | eQEP1B_in | mcasp0_fsr | mcasp0_axr3 | | uart4_rtsn | gpio0[9] |
| 34 | U4 | UART3_RTSN | lcd_data11 | gpmc_a15 | ehrpwm1B | mcasp0_ahclkr | mcasp0_axr2 | | uart3_rtsn | gpio2[17] |
| 35 | V2 | UART4_CTSN | lcd_data12 | gpmc_a16 | eQEP1A_in | mcasp0_aclkr | mcasp0_axr2 | | uart4_ctsn | gpio0[8] |
| 36 | U3 | UART3_CTSN | lcd_data10 | gpmc_a14 | ehrpwm1A | mcasp0_axr0 | | | uart3_ctsn | gpio2[16] |
| 37 | U1 | UART5_TXD | lcd_data8 | gpmc_a12 | ehrpwm1_tripzone_in | mcasp0_aclkx | uart5_txd | | uart2_ctsn | gpio2[14] |
| 38 | U2 | UART5_RXD | lcd_data9 | gpmc_a13 | ehrpwm0_synco | mcasp0_fsx | uart5_rxd | | uart2_rtsn | gpio2[15] |
| 39 | T3 | GPIO2_12 | lcd_data6 | gpmc_a6 | | eQEP2_index | | | | gpio2[12] |
| 40 | T4 | GPIO2_13 | lcd_data7 | gpmc_a7 | | eQEP2_strobe | pr1_edio_data_out7 | | | gpio2[13] |
| 41 | T1 | GPIO2_10 | lcd_data4 | gpmc_a4 | | eQEP2A_in | | | | gpio2[10] |
| 42 | T2 | GPIO2_11 | lcd_data5 | gpmc_a5 | | eQEP2B_in | | | | gpio2[11] |
| 43 | R3 | GPIO2_8 | lcd_data2 | gpmc_a2 | | ehrpwm2_tripzone_in | | | | gpio2[8] |
| 44 | R4 | GPIO2_9 | lcd_data3 | gpmc_a3 | | ehrpwm0_synco | | | | gpio2[9] |
| 45 | R1 | GPIO2_6 | lcd_data0 | gpmc_a0 | | ehrpwm2A | | | | gpio2[6] |
| 46 | R2 | GPIO2_7 | lcd_data1 | gpmc_a1 | | ehrpwm2B | | | | gpio2[7] |

### 7.1.2 Connector P9

**Table 8** lists the signals on connector **P9**. Other signals can be connected to this connector based on setting the pin mux on the processor, but this is the default settings on power up.

There are some signals that have not been listed here. Refer to the processor documentation for more information on these pins and detailed descriptions of all of the pins listed. In some cases there may not be enough signals to complete a group of signals that may be required to implement a total interface.

The **PROC** column is the pin number on the processor.
The **PIN** column is the pin number on the expansion header.
The **MODE** columns are the mode setting for each pin. Setting each mode to align with the mode column will give that function on that pin.

NOTES:

In the table are the following notations:

**PWR_BUT** is a 5V level as pulled up internally by the TPS65217C. It is activated by pulling the signal to GND.

# Both of these signals connect to pin 41 of P11. Resistors are installed that allows for the GPIO3_20 connection to be removed by removing R221. The intent is to allow the SW to use either of these signals, one or the other, on pin 41. SW should set the unused pin in input mode when using the other pin. This allowed us to get an extra signal out to the expansion header.
@ Both of these signals connect to pin 42 of P11. Resistors are installed that allows for the GPIO3_18 connection to be removed by removing R202. The intent is to allow the SW to use either of these signals, on pin 42. SW should set the unused pin in input mode when using the other pin. This allowed us to get an extra signal out to the expansion header.

## Table 8. Expansion Header P9 Pinout

| PIN | PROC | NAME | MODE0 | MODE1 | MODE2 | MODE3 | MODE4 | MODE5 | MODE6 | MODE7 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1,2 | GND | | | | | | | | | |
| 3,4 | DC_3.3V | | | | | | | | | |
| 5,6 | VDD_5V | | | | | | | | | |
| 7,8 | SYS_5V | | | | | | | | | |
| 9 | PWR_BUT | | | | | | | | | |
| 10 | A10 | SYS_RESETn | RESET_OUT | | | | | | | |
| 11 | T17 | UART4_RXD | gpmc_wait0 | mii2_crs | gpmc_csn4 | rmii2_crs_dv | mmc1_sdcd | | uart4_rxd_mux2 | gpio0[30] |
| 12 | U18 | GPIO1_28 | gpmc_be1n | mii2_col | gpmc_csn6 | mmc2_dat3 | gpmc_dir | | mcasp0_aclkr_mux3 | gpio1[28] |
| 13 | U17 | UART4_TXD | gpmc_wpn | mii2_rxerr | gpmc_csn5 | rmii2_rxerr | mmc2_sdcd | | uart4_txd_mux2 | gpio0[31] |
| 14 | U14 | EHRPWM1A | gpmc_a2 | mii2_txd3 | rgmii2_td3 | mmc2_dat1 | gpmc_a18 | | ehrpwm1A_mux1 | gpio1[18] |
| 15 | R13 | GPIO1_16 | gpmc_a0 | gmii2_txen | rmii2_tctl | mii2_txen | gpmc_a16 | | ehrpwm1_tripzone_input | gpio1[16] |
| 16 | T14 | EHRPWM1B | gpmc_a3 | mii2_txd2 | rgmii2_td2 | mmc2_dat2 | gpmc_a19 | | ehrpwm1B_mux1 | gpio1[19] |
| 17 | A16 | I2C1_SCL | spi0_cs0 | mmc2_sdwp | I2C1_SCL | ehrpwm0_synci | | | | gpio0[5] |
| 18 | B16 | I2C1_SDA | spi0_d1 | mmc1_sdwp | I2C1_SDA | ehrpwm0_tripzone | | | | gpio0[4] |
| 19 | D17 | I2C2_SCL | uart1_rtsn | timer5 | dcan0_rx | I2C2_SCL | spi1_cs1 | | | gpio0[13] |
| 20 | D18 | I2C2_SDA | uart1_ctsn | timer6 | dcan0_tx | I2C2_SDA | spi1_cs0 | | | gpio0[12] |
| 21 | B17 | UART2_TXD | spi0_d0 | uart2_txd | I2C2_SCL | ehrpwm0B | | | EMU3_mux1 | gpio0[3] |
| 22 | A17 | UART2_RXD | spi0_sclk | uart2_rxd | I2C2_SDA | ehrpwm0A | | | EMU2_mux1 | gpio0[2] |
| 23 | V14 | GPIO1_17 | gpmc_a1 | gmii2_rxdv | rgmii2_rxdv | mmc2_dat0 | gpmc_a17 | | ehrpwm0_synco | gpio1[17] |
| 24 | D15 | UART1_TXD | uart1_txd | mmc2_sdwp | dcan1_rx | I2C1_SCL | | | | gpio0[15] |
| 25 | A14 | GPIO3_21 | mcasp0_ahclkx | eQEP0_strobe | mcasp0_axr3 | mcasp1_axr1 | EMU4_mux2 | | | gpio3[21] |
| 26 | D16 | UART1_RXD | uart1_rxd | mmc1_sdwp | dcan1_tx | I2C1_SDA | | | | gpio0[14] |
| 27 | C13 | GPIO3_19 | mcasp0_fsr | eQEP0B_in | mcasp0_axr3 | mcasp1_fsx | EMU2_mux2 | | | gpio3[19] |
| 28 | C12 | SPI1_CS0 | mcasp0_ahclkr | ehrpwm0_synci | mcasp0_axr2 | spi1_cs0 | eCAP2_in_PWM2_out | | | gpio3[17] |
| 29 | B13 | SPI1_D0 | mcasp0_fsx | ehrpwm0B | | spi1_d0 | mmc1_sdcd_mux1 | | | gpio3[15] |
| 30 | D12 | SPI1_D1 | mcasp0_axr0 | ehrpwm0_tripzone | | spi1_d1 | mmc2_sdcd_mux1 | | | gpio3[16] |
| 31 | A13 | SPI1_SCLK | mcasp0_aclkx | ehrpwm0A | | spi1_sclk | mmc0_sdcd_mux1 | | | gpio3[14] |
| 32 | VADC | | | | | | | | | |
| 33 | C8 | AIN4 | | | | | | | | |
| 34 | AGND | | | | | | | | | |
| 35 | A8 | AIN6 | | | | | | | | |
| 36 | B8 | AIN5 | | | | | | | | |
| 37 | B7 | AIN2 | | | | | | | | |
| 38 | A7 | AIN3 | | | | | | | | |
| 39 | B6 | AIN0 | | | | | | | | |
| 40 | C7 | AIN1 | | | | | | | | |
| 41# | D14 | CLKOUT2 | xdma_event_intr1 | | tclkin | clkout2 | timer7_mux1 | | EMU3_mux0 | gpio0[20] |
| | D13 | GPIO3_20 | mcasp0_axr1 | eQEP0_index | | Mcasp1_axr0 | emu3 | | | gpio3[20] |
| 42@ | C18 | GPIO0_7 | eCAP0_in_PWM0_out | uart3_txd | spi1_cs1 | pr1_ecap0_ecap_capin_apwm_o | spi1_sclk | mmc0_sdwp | xdma_event_intr2 | gpio0[7] |
| | B12 | GPIO3_18 | Mcasp0_aclkr | eQEP0A_in | Mcaspo_axr2 | Mcasp1_aclkx | | | | gpio3[18] |
| 43-46 | GND | | | | | | | | | |

## 7.2 Power Jack

The DC power jack is located next to the RJ45 Ethernet connector as shown in **Figure 31**. This uses the same power connector as is used on the original BeagleBone. The connector has a 2.1mm diameter center post and a 5.5mm diameter outer dimension on the barrel.

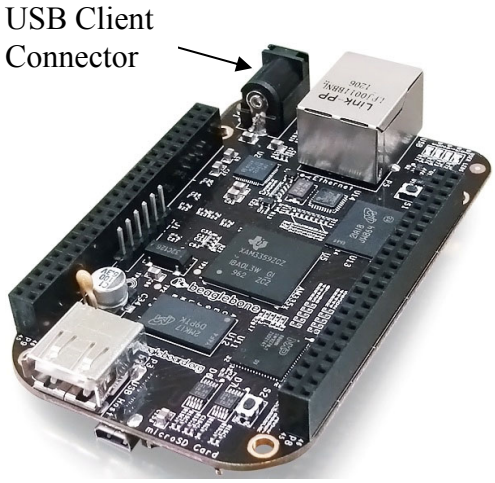


**Figure 31. 5VDC Power Jack**

The board requires a regulated 5VDC +/-.25V supply at 1A. A higher current rating may be needed if capes are plugged into the expansion headers.

## 7.3 USB Client

The USB Client connector is accessible on the bottom side of the board under the row of four LEDs as shown in **Figure 32**. It uses a 5 pin miniUSB cable, the same as is used on the original BeagleBone. The cable is provided with the board. The cable can also used to power the board.

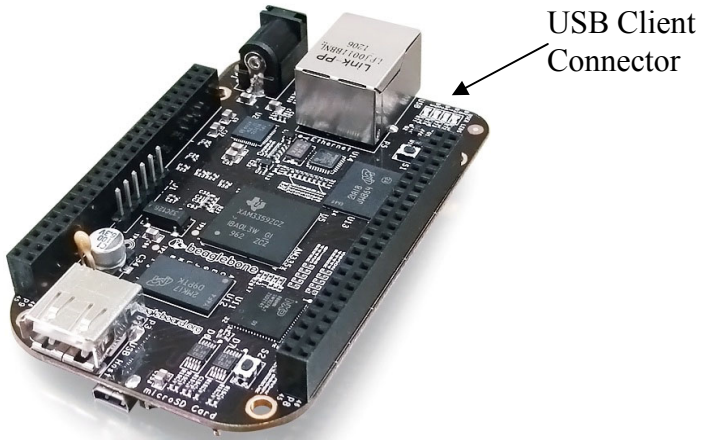


**Figure 32. USB Client Connector**

This port is a USB Client only interface.

## 7.4 USB Host

There is a single USB Host connector on the board and is shown in **Figure 33** below.

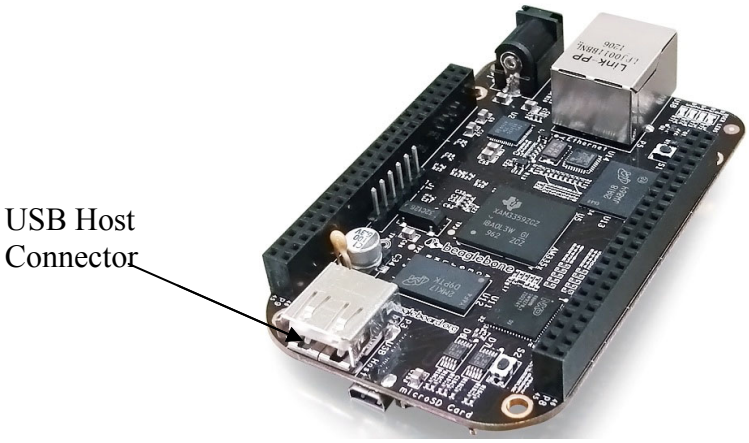


**Figure 33. USB Host Connector**

The port is USB 2.0 HS compatible and can supply up to 500mA of current.

## 7.5 Serial Header

Each board has a debug serial interface that can be accessed by using a special serial cable that is plugged into the serial header as shown in **Figure 34** below.

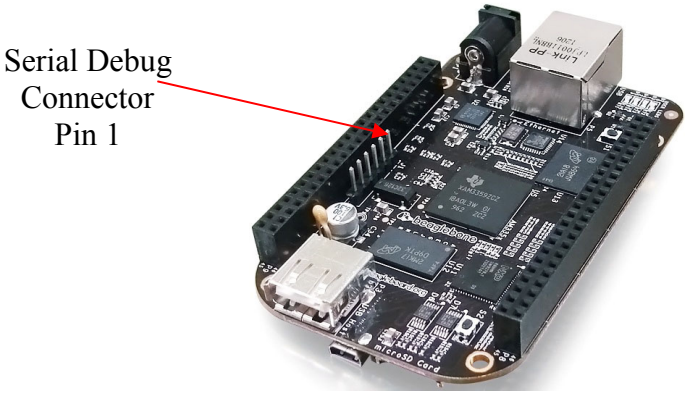


**Figure 34. Serial Debug Header**

Two signals are provided, TX and RX on this connector. The levels on these signals are 3.3V. In order access these signals a FTDI USB to Serial cable is recommended as shown in **Figure 35** below.


**Figure 35. FTDI USB to Serial Adapter**

The cable can be purchased from several different places and must be the 3.3V version TTL-232R-3V3. Information on the cable itself can be found direct from FTDI at: http://www.ftdichip.com/Support/Documents/DataSheets/Cables/DS_TTL-232R_CABLES.pdf

Pin 1 of the cable is the black wire. That must align with the pin 1 on the board which is designated by the white dot on the PCB.

## 7.6 HDMI

Access to the HDMI interface is through the HDMI connector that is located on the bottom side of the board as shown in **Figure 36** below.

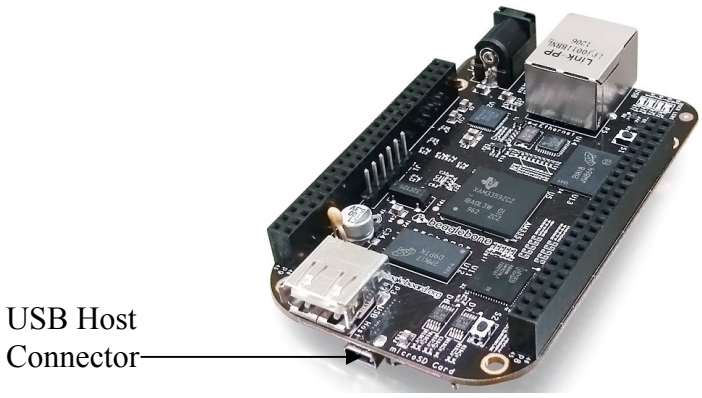


**Figure 36. HDMI Connector**

The connector is microHDMI connector. This was done due to the space limitations we had in finding a place to fit the connector. It requires a microHDMI to HDMI cable as shown in **Figure 37** below. The cable can be purchased from several different sources.

**Figure 37. HDMI Connector**

## 7.7 uSD

A microSD connector is located on the backside of the board as shown in **Figure 38** below. The SD connector is not supplied with the board.

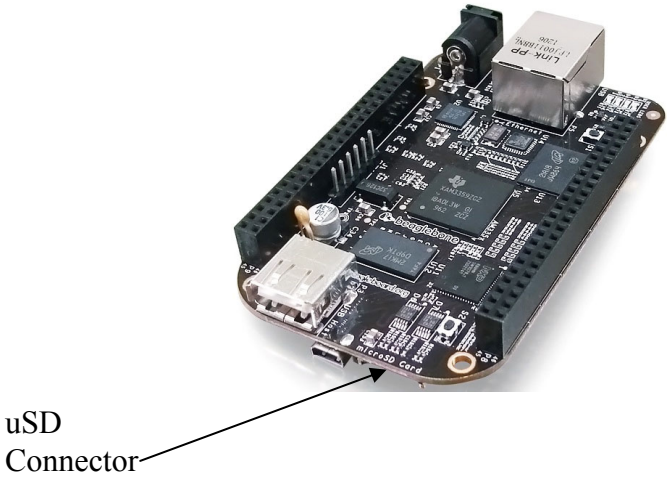


**Figure 38. uSD Connector**

When plugging in the SD card, the label should be up. Align the card with the connector and push to insert. Then release. There should be a click and the card will start to eject slightly, but it then should latch into the connector.

To eject the card, push the SD card in and then remove your finger. The SD card will be ejected from the connector. DO not try and pull the SD card out or you could damage the connector.

## 7.8 Ethernet

The board comes with a single 10/100 Ethernet interface located next to the power jack as shown in **Figure 39**.

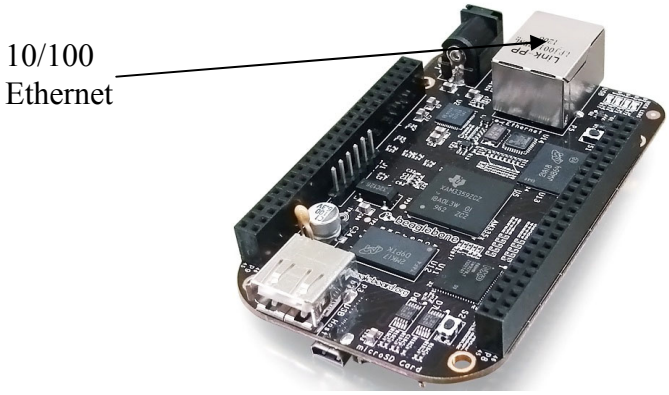


**Figure 39. Ethernet Connector**

# 8.0 Cape Board Support

The BeagleBone Black has the ability to accept up to four expansion boards or capes that can be stacked onto the expansion headers. The word cape comes from the shape of the board as it is fitted around the Ethernet connector on the main board. This notch acts as a key to insure proper orientation of the cape.

This section describes the rules for creating capes to insure proper operation with the BeagleBone Black and proper interoperability with other capes that are intended to co-exist with each other. Co-existence is not a requirement and is in itself, something that is impossible to control or administer. But, people will be able to create capes that operate with capes that are already available based on public information as it pertains to what pins and features each cape uses. This information will be able to be read from the EEPROM on each cape.

This section is intended as a guideline for those wanting to create their own capes. Its intent is not to put limits on the creation of capes and what they can do, but to set a few basic rules that will allow the SW to administer their operation with the BeagleBone Black. For this reason there is a lot of flexibility in the specification that we hope most people will find liberating and in the spirit of Open Source Hardware. I am sure there are others that would like to see tighter control, more details, more rules and much more order to the way capes are handled.

Over time, this specification will change and be updated, so please refer to the latest version of this manual prior to designing your own capes to get the latest information.

## 8.1 EEPROM

Each cape must have its own EEPROM containing information that will allow the SW to identify the board and to configure the expansion headers pins as needed. The one exception is proto boards intended for prototyping. They may or may not have an EEPROM on them. EEPROMs are required for all capes sold in order for them operate correctly when plugged into the BeagleBone Black.

The address of the EEPROM will be set via either jumpers or a dipswitch on each expansion board. **Figure 40** below is the design of the EEPROM circuit.

The EEPROM used is the same one as is used on the BeagleBone Black, a CAT24C256. The CAT24C256 is a 256 kb Serial CMOS EEPROM, internally organized as 32,768 words of 8 bits each. It features a 64−byte page write buffer and supports the Standard (100 kHz), Fast (400 kHz) and Fast−Plus (1 MHz) I2C protocol.

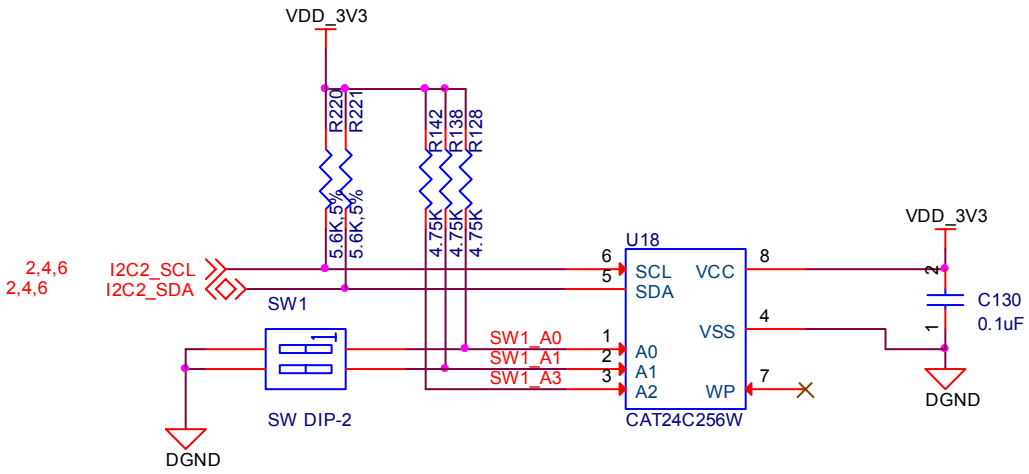


**Figure 40. Expansion Board EEPROM No Write Protect**

The addressing of this device requires two bytes for the address which is not used on smaller size EEPROMs, which only require one byte. Other compatible devices may be used as well. Make sure the device you select supports 16 bit addressing. The part package used is at the discretion of the cape designer.

### 8.1.1 EEPROM Address

In order for each cape to have a unique address, a board ID scheme is used that sets the address to be different depending on the setting of the dipswitch or jumpers on the capes. A two position dipswitch or jumpers is used to set the address pins of the EEPROM.

It is the responsibility of the user to set the proper address for each board and the position in the stack that the board occupies has nothing to do with which board gets first choice on the usage of the expansion bus signals. The process for making that determination and resolving conflicts is left up to the SW and as of this moment in time, this method is a complete mystery.

Address line A2 is always tied high. This sets the allowable address range for the expansion cards to **0x54** to **0x57**. All other I2C addresses can be used by the user in the design of their capes. But, these addresses must not be used other than for the board EEPROM information. This also allows for the inclusion of EEPROM devices on the cape if needed without interfering with this EEPROM. It requires that A2 be grounded on the EEPROM not used for cape identification.

### 8.1.2 I2C Bus

The EEPROMs on each expansion board is connected to I2C2 on connector P9 pins 19 and 20. For this reason I2C2 must always be left connected and should not be changed by

SW to remove it from the expansion header pin mux settings. If this is done, then the system will be unable to detect the capes.

The I2C signals require pullup resistors. Each board must have a 5.6K resistor on these signals. With four capes installed this will be an affective resistance of 1.4K if all capes were installed. As more capes are added the resistance is increased to overcome capacitance added to the signals. When no capes are installed the internal pullup resistors must be activated inside the processor to prevent I2C timeouts on the I2C bus.

The I2C2 bus may also be used by capes for other functions such as I/O expansion or other I2C compatible devices that do not share the same address as the cape EEPROM.

### 8.1.3 EEPROM Write Protect

The design in **Figure 40** has the write protect disabled. If the write protect is not enabled, this does expose the EEPROM to being corrupted if the I2C2 bus is used on the cape and the wrong address written to. It is recommended that a write protection function be implemented and a Test Point be added that when grounded, will allow the EEPROM to be written to. **Figure 41** shows the implementation of the EEPROM with write protect bypass enabled. Whether or not Write Protect is provided is at the discretion of the cape designer.

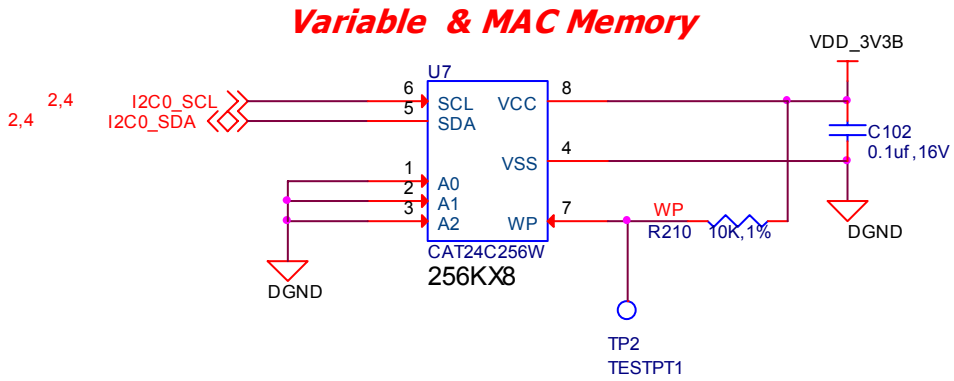


**Figure 41. Expansion Board EEPROM Write Protect**

### 8.1.4 EEPROM Data Format

**Table 9** shows the format of the contents of the expansion board EEPROM. Data is stored in Big Endian with the least significant value on the right. All addresses read single byte data from the EEPROM but are two byte addresses ASCII values are intended to be easily read by the use when the EEPROM contents are dumped.

**Table 9. Expansion Board EEPROM**

| Name | Offset | Size (bytes) | Contents |
|---|---|---|---|
| Header | 0 | 4 | 0xAA, 0x55, 0x33, 0xEE |
| EEPROM Revision | 4 | 2 | Revision number of the overall format of this EEPROM in ASCII =A1 |
| Board Name | 6 | 32 | Name of board in ASCII so user can read it when the EEPROM is dumped. Up to developer of the board as to what they call the board.. |
| Version | 38 | 4 | Hardware version code for board in ASCII. Version format is up to the developer. i.e. 02.1…00A1....10A0 |
| Manufacturer | 42 | 16 | ASCII name of the manufacturer. Company or individual's name. |
| Part Number | 58 | 16 | ASCII Characters for the part number. Up to maker of the board. |
| Number of Pins | 74 | 2 | Number of pins used by the daughter board including the power pins used. Decimal value of total pins 92 max, stored in HEX. |
| Serial Number | 76 | 12 | Serial number of the board. This is a 12 character string which is:<br>WWYY&&&&nnnn<br>where: WW = 2 digit week of the year of production<br><br>YY = 2 digit year of production<br><br>&&&&=Assembly code to let the manufacturer document the assembly number or product. A way to quickly tell from reading the serial number what the board is. Up to the developer to determine.<br><br>nnnn = incrementing board number for that week of production |
| Pin Usage | 88 | 148 | Two bytes for each configurable pins of the 74 pins on the expansion connectors<br>MSB LSB<br>Bit order: 15 14 ……1..0<br>Bit 15……..Pin is used or not……...0=Unused by cape 1=Used by cape<br>Bit 14-13……Pin Direction……..….1 0=Output 01=Input 11=BDIR<br>Bits 12-7……Reserved<br>Bit 6……..…..Slew Rate ……..0=Fast 1=Slow<br>Bit 5…….……Rx Enable……..0=Disabled 1=Enabled<br>Bit 4……..…..Pull Up/Dn Select……..0=Pulldown 1=PullUp<br>Bit 3……..Pull Up/DN enabled……..0=Enabled 1=Disabled<br>Bits 2-0 ……Mux Mode Selection……..Mode 0-7 |
| VDD_3V3EXP Current | 236 | 2 | Maximum current in milliamps. This is HEX value of the current in decimal 1500mA=0x05 0xDC 325mA=0x01 0x45 |
| VDD_5V Current | 238 | 2 | Maximum current in milliamps. This is HEX value of the current in decimal 1500mA=0x05 0xDC 325mA=0x01 0x45 |
| SYS_5V Current | 240 | 2 | Maximum current in milliamps. This is HEX value of the current in decimal 1500mA=0x05 0xDC 325mA=0x01 0x45 |
| DC Supplied | 242 | 2 | Indicates whether or not the board is supplying voltage on the VDD_5V rail and the current rating 000=No 1-0xFFFF is the current supplied storing the decimal equivalent in HEX format |
| Available | 244 | 32543 | Available space for other non-volatile codes/data to be used as needed by the manufacturer or SW driver. Could also store presets for use by SW. |

### 8.1.5 Pin Usage

**Table 10** is the locations in the EEPROM to set the I/O pin usage for the cape. It contains the value to be written to the Pad Control Registers. Details on this can be found in section **9.2.2** of the **AM335x Technical Reference Manual**, The table is left blank as a convenience and can be printed out and used as a template for creating a custom setting for each cape. The 16 bit integers and all 16 bit fields are to be stored in Big Endian format.

**Bit 15 PIN USAGE** is an indicator and should be a 1 if the pin is used or 0 if it is unused.
**Bits 14-7 RSERVED** is not to be used and left as 0.
**Bit 6 SLEW CONTROL** 0=Fast 1=Slow
**Bit 5 RX Enabled** 0=Disabled 1=Enabled
**Bit 4 PU/PD** 0=Pulldown 1=Pullup.
**Bit 3 PULLUP/DN** 0=Pullup/pulldown enabled 1= Pullup/pulldown disabled
**Bit 2-0 MUX MODE SELECT** Mode 0-7. (refer to TRM)

Refer to the TRM for proper settings of the pin MUX mode based on the signal selection to be used.

The **AIN0-6** pins do not have a pin mux setting, but they need to be set to indicate if each of the pins is used on the cape. Only bit 15 is used for the AIN signals..

## Table 10. EEPROM Pin Usage

| | | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Offset | Conn | Name | Pin Usage | Type | | Reserved | | | | | | SLEW | RX | PU-PD | PU/DEN | Mux Mode | | |
| 88 | P9-22 | UART2_RXD | | | | | | | | | | | | | | | | |
| 90 | P9-21 | UART2_TXD | | | | | | | | | | | | | | | | |
| 92 | P9-18 | I2C1_SDA | | | | | | | | | | | | | | | | |
| 94 | P9-17 | I2C1_SCL | | | | | | | | | | | | | | | | |
| 96 | P9-42 | GPIO0_7 | | | | | | | | | | | | | | | | |
| 98 | P8-35 | UART4_CTSN | | | | | | | | | | | | | | | | |
| 100 | P8-33 | UART4_RTSN | | | | | | | | | | | | | | | | |
| 102 | P8-31 | UART5_CTSN | | | | | | | | | | | | | | | | |
| 104 | P8-32 | UART5_RTSN | | | | | | | | | | | | | | | | |
| 106 | P9-19 | I2C2_SCL | | | | | | | | | | | | | | | | |
| 108 | P9-20 | I2C2_SDA | | | | | | | | | | | | | | | | |
| 110 | P9-26 | UART1_RXD | | | | | | | | | | | | | | | | |
| 112 | P9-24 | UART1_TXD | | | | | | | | | | | | | | | | |
| 114 | P9-41 | CLKOUT2 | | | | | | | | | | | | | | | | |
| 116 | P8-19 | EHRPWM2A | | | | | | | | | | | | | | | | |
| 118 | P8-13 | EHRPWM2B | | | | | | | | | | | | | | | | |
| 120 | P8-14 | GPIO0_26 | | | | | | | | | | | | | | | | |
| 122 | P8-17 | GPIO0_27 | | | | | | | | | | | | | | | | |
| 124 | P9-11 | UART4_RXD | | | | | | | | | | | | | | | | |
| 126 | P9-13 | UART4_TXD | | | | | | | | | | | | | | | | |
| 128 | P8-25 | GPIO1_0 | | | | | | | | | | | | | | | | |
| 130 | P8-24 | GPIO1_1 | | | | | | | | | | | | | | | | |
| 132 | P8-5 | GPIO1_2 | | | | | | | | | | | | | | | | |
| 134 | P8-6 | GPIO1_3 | | | | | | | | | | | | | | | | |
| 136 | P8-23 | GPIO1_4 | | | | | | | | | | | | | | | | |
| 138 | P8-22 | GPIO1_5 | | | | | | | | | | | | | | | | |
| 140 | P8-3 | GPIO1_6 | | | | | | | | | | | | | | | | |
| 142 | P8-4 | GPIO1_7 | | | | | | | | | | | | | | | | |
| 144 | P8-12 | GPIO1_12 | | | | | | | | | | | | | | | | |
| 146 | P8-11 | GPIO1_13 | | | | | | | | | | | | | | | | |
| 148 | P8-16 | GPIO1_14 | | | | | | | | | | | | | | | | |
| 150 | P8-15 | GPIO1_15 | | | | | | | | | | | | | | | | |
| 152 | P9-15 | GPIO1_16 | | | | | | | | | | | | | | | | |

| | | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Offset | Conn | Name | Pin Usage | Type | | Reserved | | | | | | SLEW | RX | PU-PD | PU/DEN | Mux Mode | | |
| 154 | P9-23 | GPIO1_17 | | | | | | | | | | | | | | | | |
| 156 | P9-14 | EHRPWM1A | | | | | | | | | | | | | | | | |
| 158 | P9-16 | EHRPWM1B | | | | | | | | | | | | | | | | |
| 160 | P9-12 | GPIO1_28 | | | | | | | | | | | | | | | | |
| 162 | P8-26 | GPIO1_29 | | | | | | | | | | | | | | | | |
| 164 | P8-21 | GPIO1_30 | | | | | | | | | | | | | | | | |
| 166 | P8-20 | GPIO1_31 | | | | | | | | | | | | | | | | |
| 168 | P8-18 | GPIO2_1 | | | | | | | | | | | | | | | | |
| 170 | P8-7 | TIMER4 | | | | | | | | | | | | | | | | |
| 172 | P8-9 | TIMER5 | | | | | | | | | | | | | | | | |
| 174 | P8-10 | TIMER6 | | | | | | | | | | | | | | | | |
| 176 | P8-8 | TIMER7 | | | | | | | | | | | | | | | | |
| 178 | P8-45 | GPIO2_6 | | | | | | | | | | | | | | | | |
| 180 | P8-46 | GPIO2_7 | | | | | | | | | | | | | | | | |
| 182 | P8-43 | GPIO2_8 | | | | | | | | | | | | | | | | |
| 184 | P8-44 | GPIO2_9 | | | | | | | | | | | | | | | | |
| 186 | P8-41 | GPIO2_10 | | | | | | | | | | | | | | | | |
| 188 | P8-42 | GPIO2_11 | | | | | | | | | | | | | | | | |
| 190 | P8-39 | GPIO2_12 | | | | | | | | | | | | | | | | |
| 192 | P8-40 | GPIO2_13 | | | | | | | | | | | | | | | | |
| 194 | P8-37 | UART5_TXD | | | | | | | | | | | | | | | | |
| 196 | P8-38 | UART5_RXD | | | | | | | | | | | | | | | | |
| 198 | P8-36 | UART3_CTSN | | | | | | | | | | | | | | | | |
| 200 | P8-34 | UART3_RTSN | | | | | | | | | | | | | | | | |
| 202 | P8-27 | GPIO2_22 | | | | | | | | | | | | | | | | |
| 204 | P8-29 | GPIO2_23 | | | | | | | | | | | | | | | | |
| 206 | P8-28 | GPIO2_24 | | | | | | | | | | | | | | | | |
| 208 | P8-30 | GPIO2_25 | | | | | | | | | | | | | | | | |
| 210 | P9-29 | SPI1_D0 | | | | | | | | | | | | | | | | |
| 212 | P9-30 | SPI1_D1 | | | | | | | | | | | | | | | | |
| 214 | P9-28 | SPI1_CS0 | | | | | | | | | | | | | | | | |
| 216 | P9-27 | GPIO3_19 | | | | | | | | | | | | | | | | |
| 218 | P9-31 | SPI1_SCLK | | | | | | | | | | | | | | | | |
| 220 | P9-25 | GPIO3_21 | | | | | | | | | | | | | | | | |

| | | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Offset | Conn | Name | Pin Usage | Type | | Reserved | | | | | | SLEW | RX | PU-PD | PU/DEN | Mux Mode | | |
| | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 222 | P8-39 | AIN0 | | | | | | | | | | | | | | | | |
| 224 | P8-40 | AIN1 | | | | | | | | | | | | | | | | |
| 226 | P8-37 | AIN2 | | | | | | | | | | | | | | | | |
| 228 | P8-38 | AIN3 | | | | | | | | | | | | | | | | |
| 230 | P9-33 | AIN4 | | | | | | | | | | | | | | | | |
| 232 | P8-36 | AIN5 | | | | | | | | | | | | | | | | |
| 234 | P9-35 | AIN6 | | | | | | | | | | | | | | | | |

## 8.2 Pin Usage Consideration

This section covers things to watch for when hooking up to certain pins on the expansion headers.

### 8.2.1 Boot Pins

There are 16 pins that control the boot mode of the processor that are exposed on the expansion headers. **Figure 42** below shows those signals:

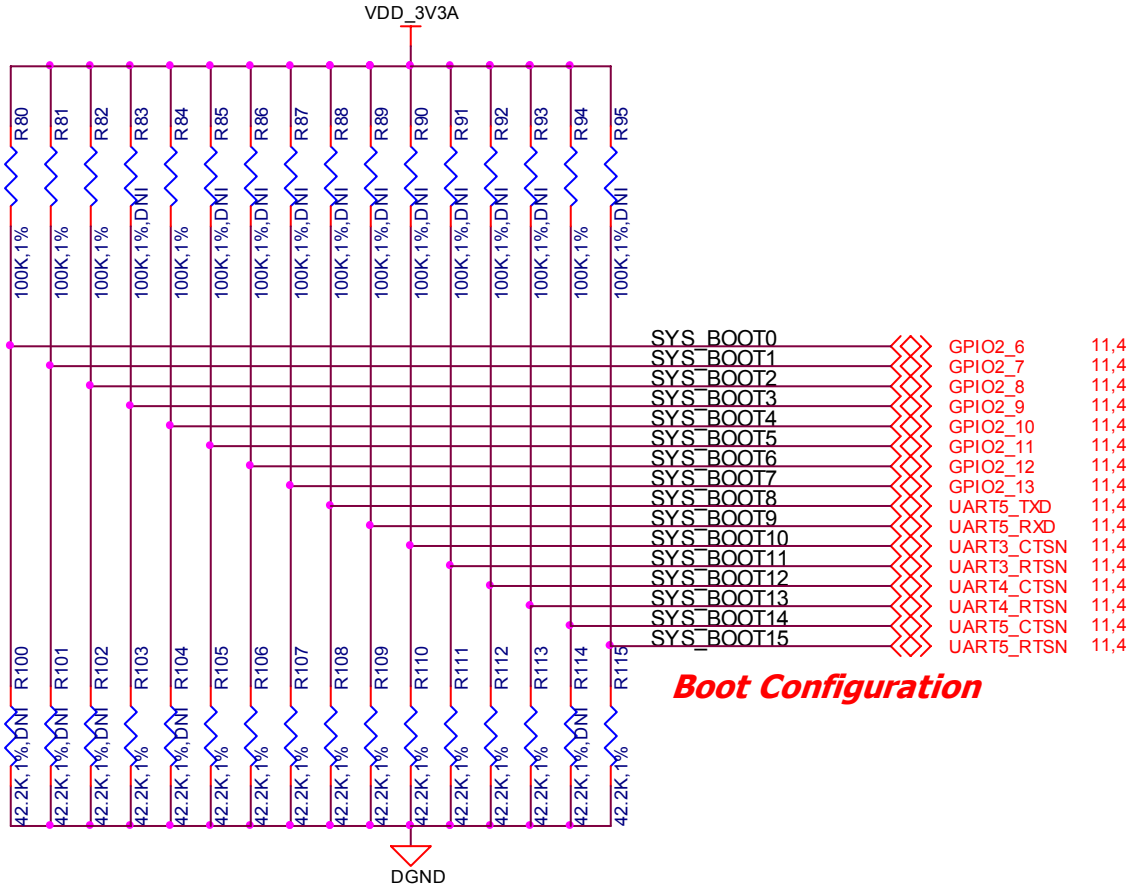


**Figure 42. Expansion Boot Pins**

If you plan to use any of these signals, then on power up, these pins should not be driven. If you do, it can affect the boot mode of the processor and could keep the processor from booting or working correctly.

If you are designing a cape that is intended to be used as a boot source, such as a NAND board, then you should drive the pins to reconfigure the boot mode, but only at reset. After the reset phase, the signals should not be driven to allow them to be used for the

other functions found on those pins. You will need to override the resistor values in order to change the settings. The DC pull-up requirement should be based on the AM335x Vih min voltage 2 volts and AM335x maximum input leakage current of 18uA when plus any other current leakage paths on these signals which you would be providing on your cape design. .

The DC pull-down requirement should be based on the AM335x Vil max voltage of 0.8 volts and AM335x maximum input leakage current of 18uA plus any other current leakage paths on these signals.

## 8.3 Expansion Connectors

A combination of male and female headers is used for access to the expansion headers on the main board. There are three possible mounting configurations for the expansion headers:

- Single-no board stacking but can be used on the top of the stack.
- Stacking-up to four boards can be stacked on top of each other.
- Stacking with signal stealing-up to three boards can be stacked on top of each other, but certain boards will not pass on the signals they are using to prevent signal loading or use by other cards in the stack.

The following sections describe how the connectors are to be implemented and used for each of the different configurations.

### 8.3.1 Non-Stacking Headers-Single Cape

For non-stacking capes single configurations or where the cape can be the last board on the stack, the two 46 pin expansion headers use the same connectors. **Figure 43** is a picture of the connector. These are dual row 23 position 2.54mm x 2.54mm connectors.

**Figure 43. Single Expansion Connector**

The connector is typically mounted on the bottom side of the board as shown in **Figure 33**. These are very common connectors and should be easily located. You can also use two single row 23 pin headers for each of the dual row headers.

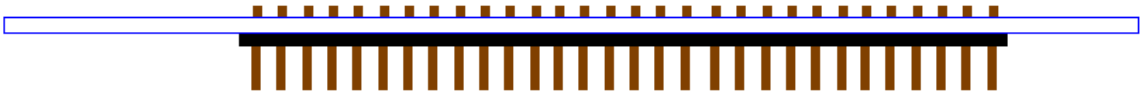

**Figure 44. Single Cape Expansion Connector**

It is allowed to only populate the pins you need. As this is a non-stacking configuration, there is no need for all headers to be populated. This can also reduce the overall cost of the cape. This decision is up to the cape designer.

For convenience listed in **Table 11** are some possible choices for part numbers on this connector. They have varying pin lengths and some may be more suitable than others for your use. It should be noted, that the longer the pin and the further it is inserted into the BeagleBone Black connector, the harder it will be to remove due to the tension on 92 pins. This can be minimized by using shorter pins or removing those pins that are not used by your particular design. The first item in **Table 16** is on the edge and may not be the best solution. Overhang is the amount of the pin that goes past the contact point of the connector on the BeagleBone Black.

Refer to **Section 8.3** for more information on the connectors and the insertion force issue.

**Table 11. Single Cape Connectors**

| SUPPLIER | PARTNUMBER | TAIL LENGTH(in) | OVERHANG(in) |
|---|---|---|---|
| Major League | TSHC-123-D-03-145-GT-LF | .145 | .004 |
| Major League | TSHC-123-D-03-240-GT-LF | .240 | .099 |
| Major League | TSHC-123-D-03-255-GT-LF | .255 | .114 |

The GT in the part number is a plating option. Other options may be used as well as long as the contact area is gold. Other possible sources are Sullins and Samtec for these connectors. You will need to insure the depth into the connector is sufficient

### 8.3.2 Battery Connector- Single

For non-stacking or single configuration this connector is a single 10 pin expansion header. **Figure 45** is a picture of the connector. This is a dual row 10 position 2.54mm x 2.54mm connectors. This is the same connector as the main connectors, only shorter.

**Figure 45. Battery/Backlight Expansion Connector**

**Table 12** is the possible part numbers for this connector. The first item in **Table 12** is on the edge and may not be the best solution. Overhang is the amount of the pin that goes past the contact point of the connector on the BeagleBone Black.

Refer to **Section 8.3** for more information on the connectors and the insertion force issue.

**Table 12. Single Cape Backlight Connectors**

| SUPPLIER | PARTNUMBER | TAIL LENGTH(in) | OVERHANG(in) |
|---|---|---|---|
| Major League | TSHC-105-D-03-145-GT-LF | .145 | .004 |
| Major League | TSHC-105-D-03-240-GT-LF | .240 | .099 |
| Major League | TSHC-105-D-03-255-GT-LF | .255 | .114 |

### 8.3.3 Main Expansion Headers-Stacking

For stacking configuration, the two 46 pin expansion headers use the same connectors. **Figure 46** is a picture of the connector. These are dual row 23 position 2.54mm x 2.54mm connectors.

**Figure 46. Expansion Connector**

The connector is mounted on the top side of the board with longer tails to allow insertion into the BeagleBone Black. **Figure 47** is the connector configuration for the connector.

**Figure 47. Stacked Cape Expansion Connector**

For convenience listed in **Table 18** are some possible choices for part numbers on this connector. They have varying pin lengths and some may be more suitable than others for your use. It should be noted, that the longer the pin and the further it is inserted into the

BeagleBone Black connector, the harder it will be to remove due to the tension on 92 pins. This can be minimized by using shorter pins. There are most likely other suppliers out there that will work for this connector as well. If anyone finds other suppliers of compatible connectors that work, let us know and they will be added to this document. The first item in **Table 13** is on the edge and may not be the best solution. Overhang is the amount of the pin that goes past the contact point of the connector on the BeagleBone Black.

Please refer to **Section 8.3** for more information on the connectors and the insertion force issue. The third part listed in **Table 13** will have insertion force issues.

**Table 13. Stacked Cape Connectors**

| SUPPLIER | PARTNUMBER | TAIL LENGTH(in) | OVERHANG(mm) |
|---|---|---|---|
| Major League | SSHQ-123-D-06-GT-LF | .190 | 0.049 |
| Major League | SSHQ-123-D-08-GT-LF | .390 | 0.249 |
| Major League | SSHQ-123-D-10-GT-LF | .560 | 0.419 |

There are also different plating options on each of the connectors above. Gold plating on the contacts is the minimum requirement. If you choose to use a different part number for plating or availability purposes, make sure you do not select the "LT" option.
Other possible sources are Sullins and Samtec but make sure you select one that has the correct mating depth.

### 8.3.4 Stacked Capes w/Signal Stealing

**Figure 38** is the connector configuration for stackable capes that does not provide all of the signals upwards for use by other boards. This is useful if there is an expectation that other boards could interfere with the operation of your board by exposing those signals for expansion. This configuration consists of a combination of the stacking and non-stacking style connectors.

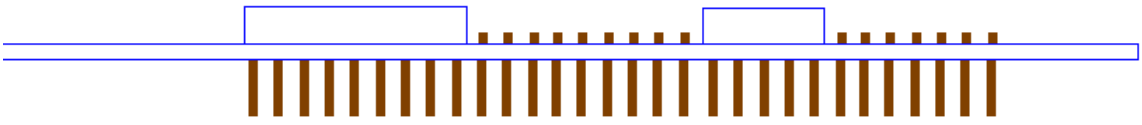

**Figure 48. Stacked w/Signal Stealing Expansion Connector**

### 8.3.5 Retention Force

The length of the pins on the expansion header has a direct relationship to the amount of force that is used to remove a cape from the BeagleBone Black. The longer the pins

extend into the connector the harder it is to remove. There is no rule that says that if longer pins are used, that the connector pins have to extend all the way into the mating connector on the BeagleBone Black, but this is controlled by the user and therefore is hard to control.

This section will attempt to describe the tradeoffs and things to consider when selecting a connector and its pin length.

### 8.3.6 BeagleBone Black Female Connectors

**Figure 29** shows the key measurements used in calculating how much the pin extends past the contact point on the connector, what we call overhang.

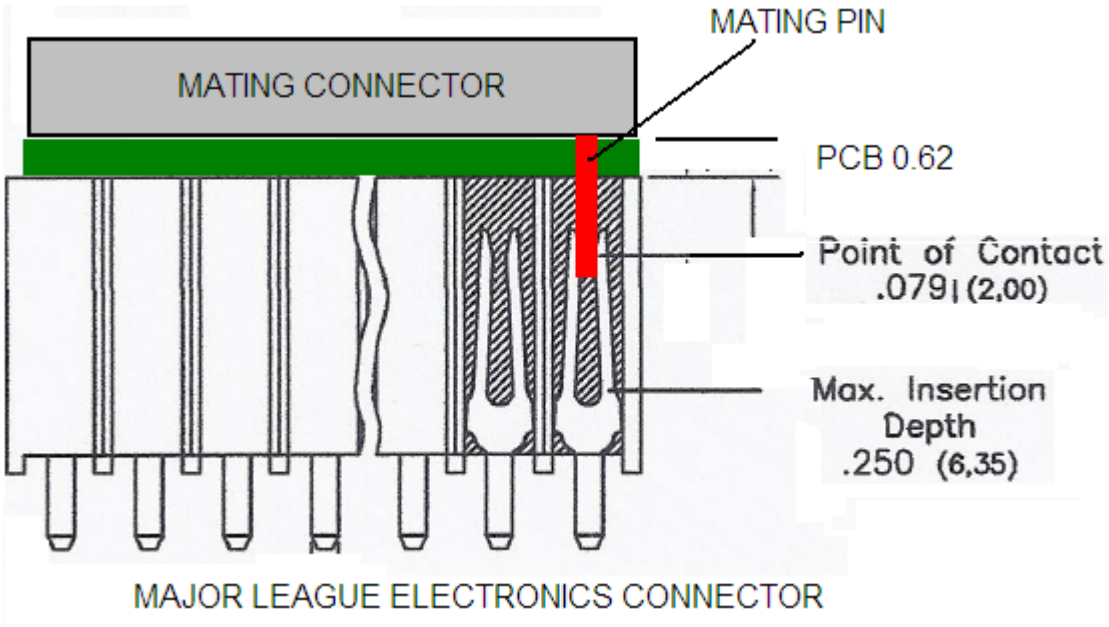


**Figure 49. Connector Pin Insertion Depth**

To calculate the amount of the pin that extends past the Point of Contact, use the following formula:

Overhang=Total Pin Length- PCB thickness (.062) - contact point (.079)

The longer the pin extends past the contact point, the more force it will take to insert and remove the board. Removal is a greater issue than the insertion.

## 8.4 Signal Usage

Based on the pin muxing capabilities of the processor, each expansion pin can be configured for different functions. When in the stacking mode, it will be up to the user to insure that any conflicts are resolved between multiple stacked cards. When stacked, the first card detected will be used to set the pin muxing of each pin. This will prevent other modes from being supported on stacked cards and may result in them being inoperative.

In **Section 7.12** of this document, the functions of the pins are defined as well as the pin muxing options. Refer to this section for more information on what each pin is. To simplify things, if you use the default name as the function for each pin and use those functions, it will simplify board design and reduce conflicts with other boards.

Interoperability is up to the board suppliers and the user. This specification does not specify a fixed function on any pin and any pin can be used to the full extent of the functionality of that pin as enabled by the processor.

## 8.5 Cape Power

This section describes the power rails for the capes and their usage.

### 8.5.1 Main Board Power

The **Table 14** describes the voltages from the main board that are available on the expansion connectors and their ratings. All voltages are supplied by connector **P9**. The current ratings listed are per pin.

**Table 14. Expansion Voltages**

| Current | Name | P9 | | Name | Current |
|---|---|---|---|---|---|
| | GND | 1 | 2 | GND | |
| 250mA | VDD_3V3EXP | 3 | 4 | VDD_3V3EXP | 250mA |
| 1000mA | VDD_5V | 5 | 6 | VDD_5V | 1000mA |
| 250mA | SYS_5V | 7 | 8 | SYS_5V | 250mA |
| | | : | : | | |
| | GND | 43 | 44 | GND | |
| | GND | 45 | 46 | GND | |

The **VDD_3V3EXP** rail is supplied by the LDO on the BeagleBone Black and is the primary power rail for expansion boards. If the power requirement for the capes exceeds the current rating, then locally generated voltage rail can be used. It is recommended that this rail be used to power any buffers or level translators that may be used.

**VDD_5V** is the main power supply from the DC input jack. This voltage is not present when the board is powered via USB. The amount of current supplied by this rail is dependent upon the amount of current available. Based on the board design, this rail is limited to 1A per pin from the main board.

The **SYS_5V** rail is the main rail for the regulators on the main board. When powered from a DC supply or USB, this rail will be 5V. The available current from this rail depends on the current available from the USB and DC external supplies.

### 8.5.2 Expansion Board External Power

A cape can have a jack or terminals to bring in whatever voltages may be needed by that board. Care should be taken not to let this voltage feedback into any of the expansion header pins.

It is possible to provide 5V to the main board from an expansion board. By supplying a 5V signal into the **VDD_5V** rail, the main board can be supplied. This voltage must not exceed 5V. You should not supply any voltage into any other pin of the expansion connectors. Based on the board design, this rail is limited to 1A per pin to the BeagleBone Black.

## 8.6 Mechanical

This section provides the guidelines for the creation of expansion boards from a mechanical standpoint. Defined is a standard board size that is the same profile as the BeagleBone Black. It is expected that the majority of expansion boards created will be of standard size. It is possible to create boards of other sizes and in some cases this is required, as in the case of an LCD larger than the BeagleBone Black board.

### 8.6.1 Standard Cape Size

**Figure 50** is the outline of the standard cape. The dimensions are in inches.

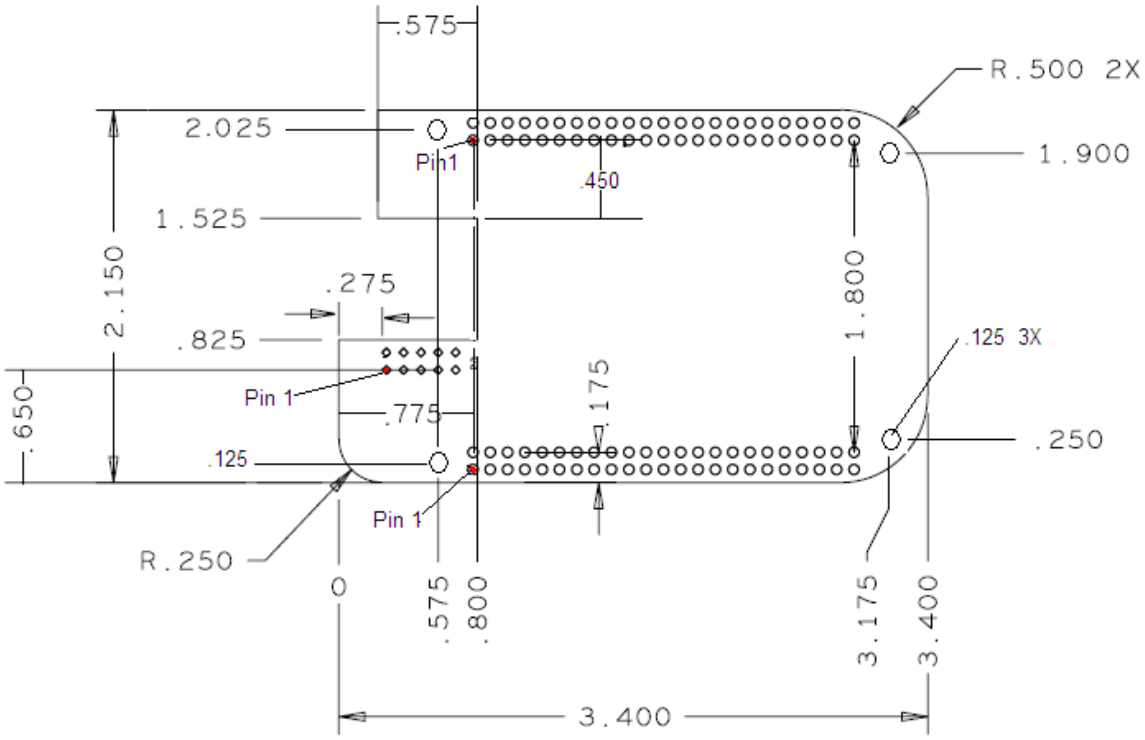


**Figure 50. Cape Board Dimensions**

A slot is provided for the Ethernet connector to stick up higher than the cape when mounted. This also acts as a key function to insure that the cape is oriented correctly. Space is also provided to allow access to the user LEDs and reset button on the main board.
Some people have inquired as to the difference in the radius of the corners of the BeagleBone Black and why they are different. This is a result of having the BeagleBone fit into the Altoids style tin.

It is not required that the cape be exactly like the BeagleBone Black board in this respect.

### 8.6.2 Extended Cape Size

Capes larger than the standard board size are also allowed. A good example would be an LCD panel. There is no practical limit to the sizes of these types of boards. The notch for the key is also not required, but it is up to the supplier of these boards to insure that the BeagleBone Black is not plugged in incorrectly in such a manner that damage would be cause to the BeagleBone Black or any other capes that may be installed. Any such damage will be the responsibility of the supplier of such a cape to repair.

As with all capes, the EEPROM is required and compliance with the power requirements must be adhered to.

### 8.6.3 Enclosures

There are numerous enclosures being created in all different sizes and styles. The mechanical design of these enclosures is not being defined by this specification.

The ability of these designs to handle all shapes and sizes of capes, especially when you consider up to four can be mounted with all sorts of interface connectors, it is difficult to define a standard enclosure that will handle all capes already made and those yet to be defined.

If cape designers want to work together and align with one enclosure and work around it that is certainly acceptable. But we will not pick winners and we will not do anything that impedes the openness of the platform and the ability of enclosure designers and cape designers to innovate and create new concepts.

# 9.0 BeagleBone Black Mechanical Specification

| | |
|---|---|
| Size: | 3.5” x 2.15” (86.36mm x 53.34mm) |
| Max height: | .187” (4.76mm) |
| PCB Layers: | 6 |
| PCB thickness: | .062” |
| RoHS Compliant: | Yes |
| Weight: | 1.4 oz |

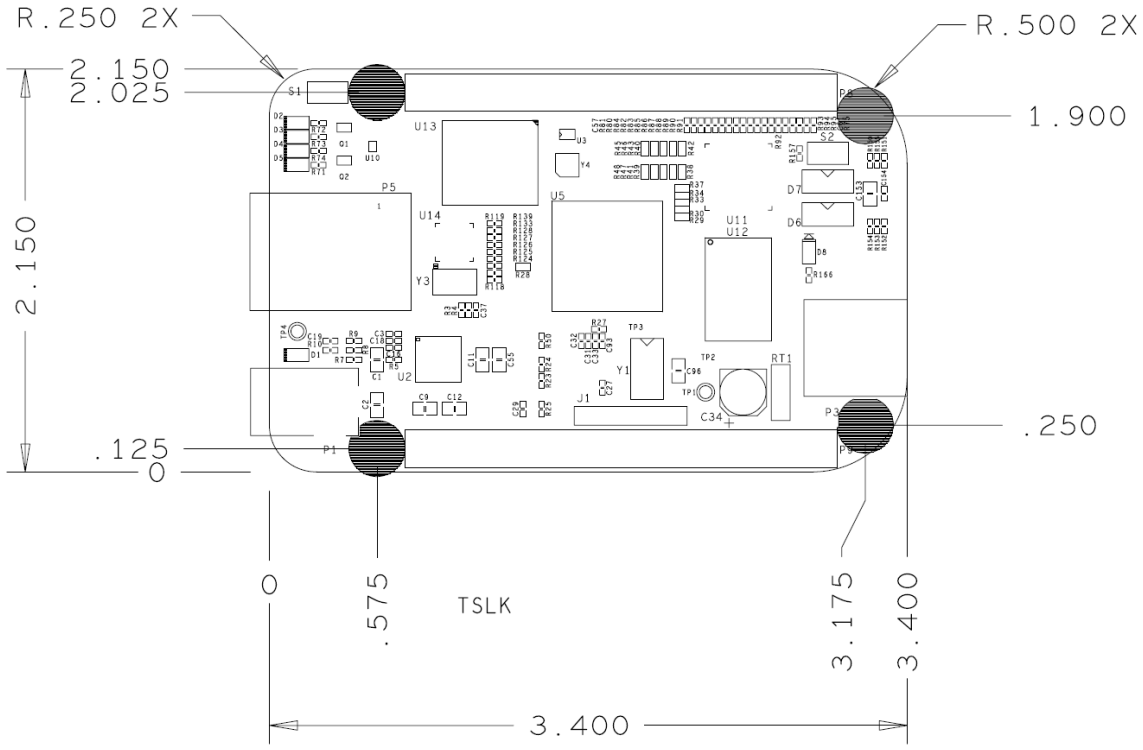


**Figure 51. Board Top Side Profile**

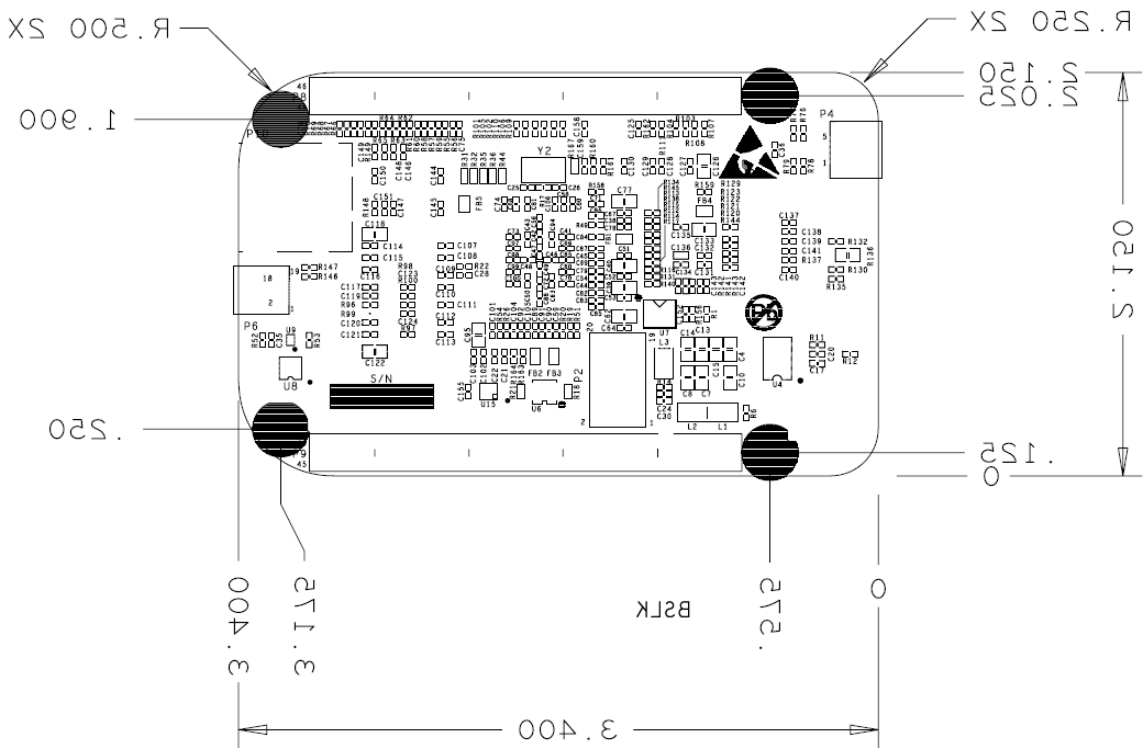


**Figure 52. Board Bottom Profile**

## 10.0 Design Information

Design information can be found on the SD card that ships with board under the documents/hardware directory when connected over the USB cable. Provided there is:

- Schematic in PDF
- Schematic in OrCAD (Cadence Design Entry CIS 16.3)
- PCB Gerber
- PCB Layout File (Allegro)
- Bill of Material
- System Reference Manual (This document).

You can also download the files from http://beagleboard.org/hardware/design or from the Circuitco WIKI at http://circuitco.com/support/index.php?title=BeagleBoneBlack

ALL support for this design is through the BeagleBoard.org community at beagleboard@googlegroups.com or http://beagleboard.org/discuss .