BeagleBone Rev A6
System Reference Manual

Revision 0.0

May 9, 2012

Send all comments and errors to the author:

Gerald Coley gerald@beagleboard.org

NOTE: This device has been tested and verified to comply with Part 15, Class B, of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

NOTE: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Changes or modifications not expressly approved by this manual for compliance could void the user's authority to operate the equipment.

THIS DOCUMENT

For any questions, concerns, or issues submit them to gerald@BeagleBoard.org

BEAGLEBONE DESIGN

These design materials referred to in this document are ***NOT SUPPORTED*** and **DO NOT** constitute a reference design. Only "community" support is allowed via resources at BeagleBoard.org/discuss.

THERE IS NO WARRANTY FOR THE DESIGN MATERIALS, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE DESIGN MATERIALS "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE DESIGN MATERIALS IS WITH YOU. SHOULD THE DESIGN MATERIALS PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

We mean it; these design materials may be totally unsuitable for any purposes.

BeagleBoard.org provides the enclosed product(s) under the following conditions:

This evaluation board/kit is intended for use for **ENGINEERING DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY** and is not considered by BeagleBoard.org to be a finished end-product fit for general consumer use. Persons handling the product(s) must have electronics training and observe good engineering practice standards. As such, the goods being provided are not intended to be complete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety and environmental measures typically found in end products that incorporate such semiconductor components or circuit boards. This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL, and therefore may not meet the technical requirements of these directives or other related directives.

Should this evaluation board/kit not meet the specifications indicated in the User's Guide, the board/kit may be returned within **30 days** from the date of delivery for a full refund to the distributor form which you purchased the board. THE FOREGOING WARRANTY IS THE EXCLUSIVE WARRANTY MADE BY SELLER TO BUYER AND IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE.

The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies BeagleBoard.org from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge.

EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER PARTY SHALL BE LIABLE TO THE OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

BeagleBoard.org currently deals with a variety of customers for products, and therefore our arrangement with the user **is not exclusive**. BeagleBoard.org assumes **no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein**.

Please read the System Reference Manual and, specifically, the Warnings and Restrictions notice in the User's Guide prior to handling the product. This notice contains important safety information about temperatures and voltages. For additional information on BeagleBoard.org environmental and/or safety programs, please visit BeagleBoard.org.

No license is granted under any patent right or other intellectual property right of BeagleBoard.org covering or relating to any machine, process, or combination in which such BeagleBoard.org products or services might be or are used.

Mailing Address:

BeagleBoard.org
1380 Presidential Dr. #100
Richardson, TX 75081
U.S.A.

WARRANTY: *The BeagleBoard is warranted against defects in materials and workmanship for a period of 90 days from purchase. This warranty does not cover any problems occurring as a result of improper use, modifications, exposure to water, excessive voltages, abuse, or accidents. All boards will be returned via standard mail if an issue is found. If no issue is found or express return is needed, the customer will pay all shipping costs*.

Before returning the board, please visit
BeagleBoard.org/support

For up to date SW images and technical information refer to
http://circuitco.com/support/index.php?title=BeagleBone

Please refer to **Section 9** of this document for the board checkout procedures.

To return a defective board, please request an RMA at http://beagleboard.org/support/rma

Please DO NOT return the board without approval from the RMA team first.

All boards received without RMA approval will not be worked on.

Table of Contents

Figures

Tables

1.0 Introduction

This document is the **System Reference Manual** for the BeagleBone. It covers revision **A3** thru **A6**. It is intended as a guide to assist anyone purchasing or who are considering purchasing the board to understand the overall system design and the features of the BeagleBone. It can also be used as a reference for the design for those who are implementing this design into their own product.

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

For support, the primary mailing list is discuss@beagleboard.org
For HW support use the mailing list and also refer to the HW support WIKI at

2.0 Change History

2.1 Change History

Table 1. Change History

Rev	Changes	Date	By
0.1	Original Release for review	November 4, 2011	GC
0.2	1. Added notch dimension to the Cape board outline. 2. Added power numbers to features table. 3. Corrected USB0 and USB1 numbering 4. Made correction on two signals on Tables 10 thru 12.	November 11, 2011	GC
A4.0.4	1. Rev A4 Release 2. Documented the changes. 3. Updated Figure 28 to show pullup resistors as 5.6K. 4. Added note to Cape section that mounting holes are not required. 5. Fixed link to the TPS65217B documentation. 6. Added section on ADC interface.	January 3, 2012	GC
A5	1. Added clarification on image creation process. 2. Added more detail on USB 5VDC supplied to Capes. 3. Corrected section 6.3.6 to reflect four UART ports instead of five. 4. Updated Figure 36 with more hole dimensions. 5. Added section on the rev A4 to Rev A5 changes. 6. Made changes in Table 12. 7. Added note on polarity of Yellow Ethernet LED in section 7.8.7.	January 31, 2012	GC
A6	1. Added changes for rev A6 that covered fixing of the link LED, JTAG Reset, and DHCP issue. 2. Added PRU information and two additional signals for the PRU. 3. Added write protection to EEPROM. 4. Updated Cape section. Added clarifications and more information. 5. Fixed numbering of subsections in Section 7.0 6. Fixed error in Table 9 pin 6 to MMC1_DAT3. 7. Fixed error in Table 9 pin 22 Mode 1 should be MMC1_DAT5 and Mode 2 is now blank. 8. Fixed error in Table 9 pin 23Mode 1 should be MMC1_DAT4. 9. Updated Table 7 to show the revision number in the EEPROM matches the revision of the board. 10. Corrected various typos. 11. Updated Battery Interface section to accurately document the LDO dropout at 200mV. 12. Added SW Support section.	May 9, 2012	GC

2.2 Rev A5 vs. A6

Rev **A6** underwent several changes:

- Fixed the Yellow Link LED and R219 issue by adding a pulldown to the SMSC PHY.
- Added two PRU signals top provide a full 8bit PRU interface when the LCD board is installed.
- Move the resistors that where too close to the standoff.
- Removed connection to the VPP pin from the layout.
- Fixed spurious reset issues on JTAG connect.
- Addressed LAN8710 default mode.

There were no changes made that affect the operation of the board form a SW perspective. Feature and operation wise the **A6** is the same as an A3.

2.2.1 PCB Changes

Here are the changes that affected the PCB:

1) Added R220
2) Added R217, R218, R202, and R221.
3) Added etch to route the PRU signals to the expansion header using the above resistors.
4) Moved R180 and R150.
5) Changed revision to C2.

2.2.2 Design Changes

1) Changed R219 is now installed.
2) Added R220 a 10K pulldown to pin 18 of the SMSC PHY to allow R219 addition to work as expected.
3) Removed the connection to the VPP pin on the processor.
4) Added R221, R218, R217, R202 to facilitate the addition of two signals, GPIO3_18 and GPIO3_19 to the expansion bus header to provide two more signals for the PRU access.
5) Changed R210 to installed and added test point to allow the EEPROM to be programmed but with added protection to prevent corruption. Also added Test Point to enable programming.
6) Moved resistors R189 and R150 to provide more clearance around mounting hole.
7) Removed R122 which was not connected to the correct pin on the on the LAN8710 for setting the HW default mode.
8) Removed R163 to disconnect the FT2232 reset out that was causing spurious resets when connecting the JTAG on a running board.
9) Added above changes as needed to the BOM.

2.3 Rev A4 vs. A5

There was a key issue with rev A4 where R219 was causing some unintended issues with the operation of the Ethernet interface.

2.3.1 PCB Changes

There were no PCB changes.

2.3.2 Design Changes

1) R219 was removed from the assembly. It was installed on Rev A6 with a PCB change.

2.3.3 Production Changes

1) Changes were made in production testing to test for bad Reset switches
2) Reset switches are not being taken through the wash.
3) The FTDI VID was changed to 0403 and the PID was changed to 6010. Description was changed to "BeagleBone/XDS100"

This version of the board returns the functionality of the board to that of the Rev **A3** via the removal of R219. It uses the same PCB revision as the A4. It also ships with an updated version of the Angstrom image providing out of the box support for the DVI-D and 7" LCD Capes.

There will be three possible versions of the Rev A5. One will be the new production version that is built from the ground up as an A5, R219 not installed.

The second version will be a reworked Revision A4 that has R219 removed at the factory and retested.

The third version will be a revision A3 that just has the updated SW added. All reworked versions will have the reset switches double checked as well. All reworked boards will be retested using the full production test process.

You will be able to identify these versions via the serial number. They all will be labeled as revision A5. The two digits after the BB in the serial number, S/N: 5111BB000023, will indicate the board. A fresh revision A5, will be 00, A4 reworked will be 01, and a recertified A3 will be 02. There is no functional or operational difference between any of these boards. They are all revision **A5** and will ship with the same SW.

For those with Revision **A3** and **A4**, you will be able to download the latest shipping image from http://circuitco.com/support/index.php?title=BeagleBone and have all the features of the Revision **A5**. For **A4** users, you will need to remove R219 and instructions are provided at http://circuitco.com/support/index.php?title=BeagleBone .

2.4 Rev A3 vs. A4

No functional changes were made to the board as it relates to its overall operation other than the LED fix for the Speed indicator on the Ethernet connector. Main change was the addition of a different SD connector.

2.4.1 PCB Changes

The following PCB changes were made to facilitate the acquisition of components to meet the production schedule which required different footprints.

- New microSD connector. PCB layout was changed to facilitate the change.
- 50 ohm resistor was changed to a 0402 footprint.
- Changes C7 footprint to 0805.

2.4.2 Design Changes

Added a 10k pull down resistor, R219, to fix polarity of the speed LED on the Ethernet connector.

NOTE: **The pictures in this document were not changed to reflect the A4/A5/A6 versions. The benefit of doing this is very small. The only obvious difference is the big resistor below the USB Host connector is no longer there.**

2.5 Known Issues

For an up to date list of all known issue per revision, please refer to the HW WIKI support page at http://circuitco.com/support/index.php?title=BeagleBone#Known_Issues
.

2.6 BeagleBone Overview

The BeagleBone 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 via onboard support of some interfaces.

2.7 BeagleBone Expansion

By utilizing comprehensive expansion connectors, the BeagleBone is highly extensible to add many features and interfaces via add-on boards or Capes. Capes refer to the shape of the add-on boards and are discussed later in this document. A majority of the signals from the processor are exposed via the expansion headers and can be accessed there, but may require additional hardware in order to use them. This will be handled by the creation of Capes in the future. Due to the deep multiplexing of the pins, there are limits as to how many interfaces can coexist at any one time. Refer to the processor documentation for more information.

2.8 BeagleBone Design Material

All of the design information is freely available and can be used as the basis for a product or design. If the user decides to use the BeagleBone design in a product, they assume all responsibility for such use and are totally responsible for all aspects of its use.

We do not sell BeagleBone boards for use in end products. We choose to utilize our resources to create boards for the expressed purpose as previously stated. We will be changing the design to improve it and will not continue to make older revisions as the overall design matures.

There are programs available for someone to have the board built to their specifications and then use that board in a product. All of the design information is freely available and will be kept up to date. Anyone is free to use that information as previously stated.

2.9 In The Box

The BeagleBone ships in a box with the following components:

- BeagleBone
- USB Cable
- 4GB uSD card with SW and documentation

3.0 BeagleBone Features and Specification

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

Table 2 provides a list of the BeagleBone's features.

Table 2. BeagleBone Features

	Feature	
Processor	AM3359 500MHZ-USB Powered 720MHZ-DC Powered	
Memory	256MB DDR2 400MHZ (128MB Optional)	
PMIC TPS65217B	Power Regulators	
	LiION Single cell battery charger (via expansion*)	
	20mA LED Backlight driver, 39V, PWM (via expansion*)	
	*(Additional components required)	
Debug Support	USB to Serial Adapter	miniUSB connector
	On Board JTAG via USB	4 USER LEDs
		Optional 20-pin CTI JTAG
Power	USB	5VDC External jack
PCB	3.4" x 2.1"	6 layers
Indicators	Power	
	4-User Controllable LEDs	
HS USB 2.0 Client Port	Access to the USB1 Client mode	
HS USB 2.0 Host Port	USB Type A Socket, 500mA LS/FS/HS	
Ethernet	10/100, RJ45	
SD/MMC Connector	microSD , 3.3V	
User Interface	1-Reset Button	
Overvoltage Protection	Shutdown @ 5.6V MAX	
Expansion Connectors	Power 5V, 3.3V , VDD_ADC(1.8V) 3.3V I/O on all signals 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, Battery Charger, LED Backlight, Expansion Board ID (Up to 3 can be stacked)	
5V Power	USB or 5.0VDC to 5.2VDC See Table 3 for power consumption numbers.	
Weight	1.4 oz (39.68 grams)	

*Board will boot to 500MHz under USB power.

NOTE: DUE TO MULIPLEXING ON THE PINS OF THE PROCESSOR, ALL OF THESE EXPANSION SIGNALS CANNOT BE AVAILABLE AT THE SAME TIME.

NOTE: The battery configuration is not suitable to power the BeagleBone in its current configuration.

The following sections provide more detail on each feature and are covered under each section of this document.

3.1 Board Component Locations

The **Figure 1** below shows the top side locations of the key components on the PCB layout of the BeagleBone.

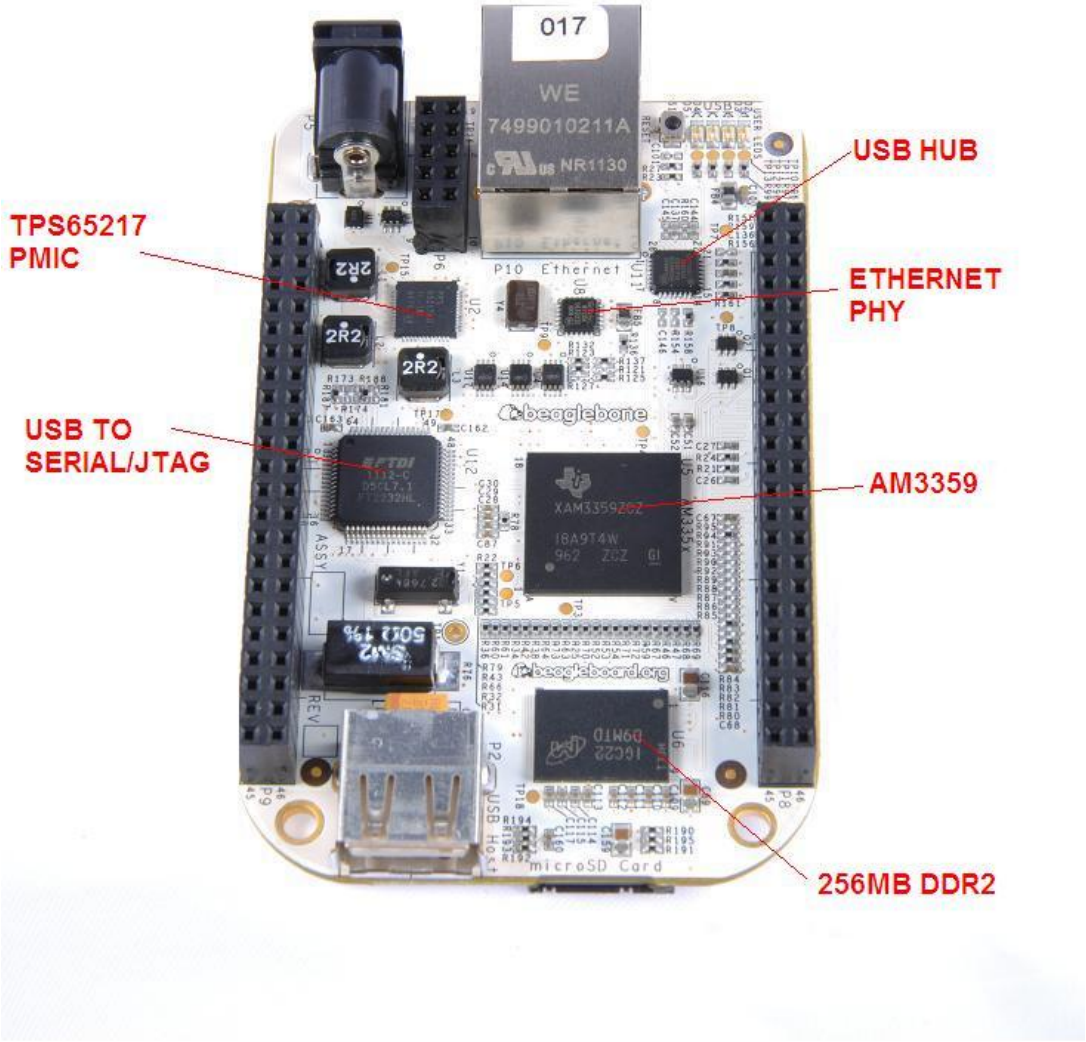
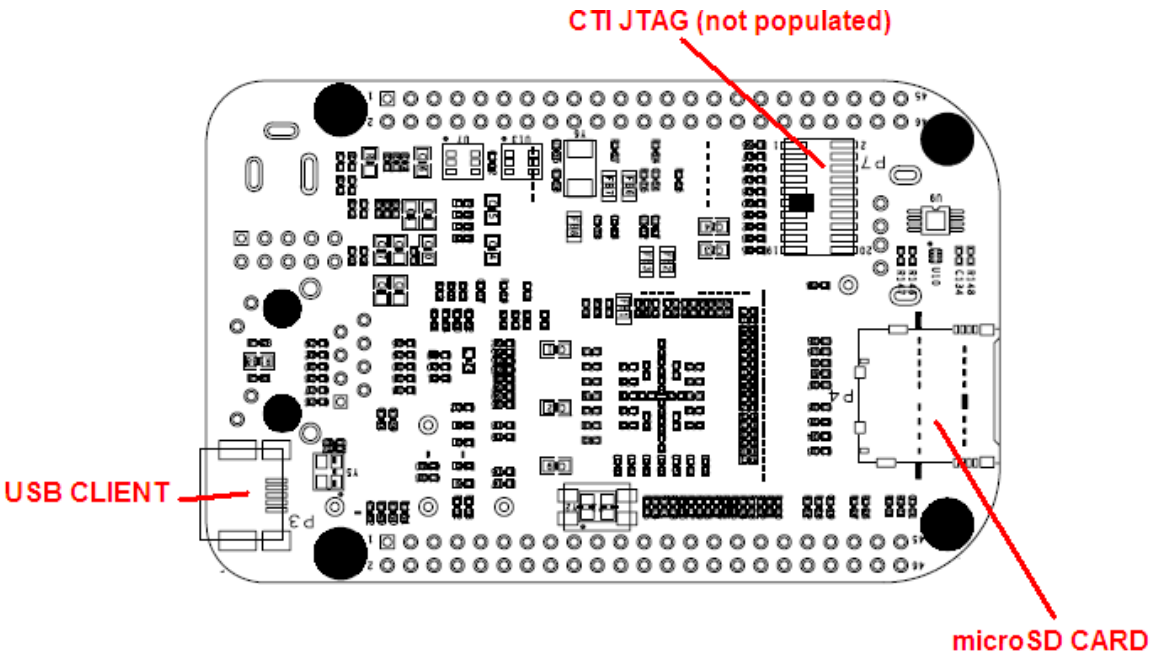


Figure 1. Top Side Components

Figure 2 shows the key components mounted on the back side of the board.



7

Figure 2. Bottom Side Components

3.2 Board Connector and Indicator Locations

Figure 3 shows the key connector and LED locations of the BeagleBone.

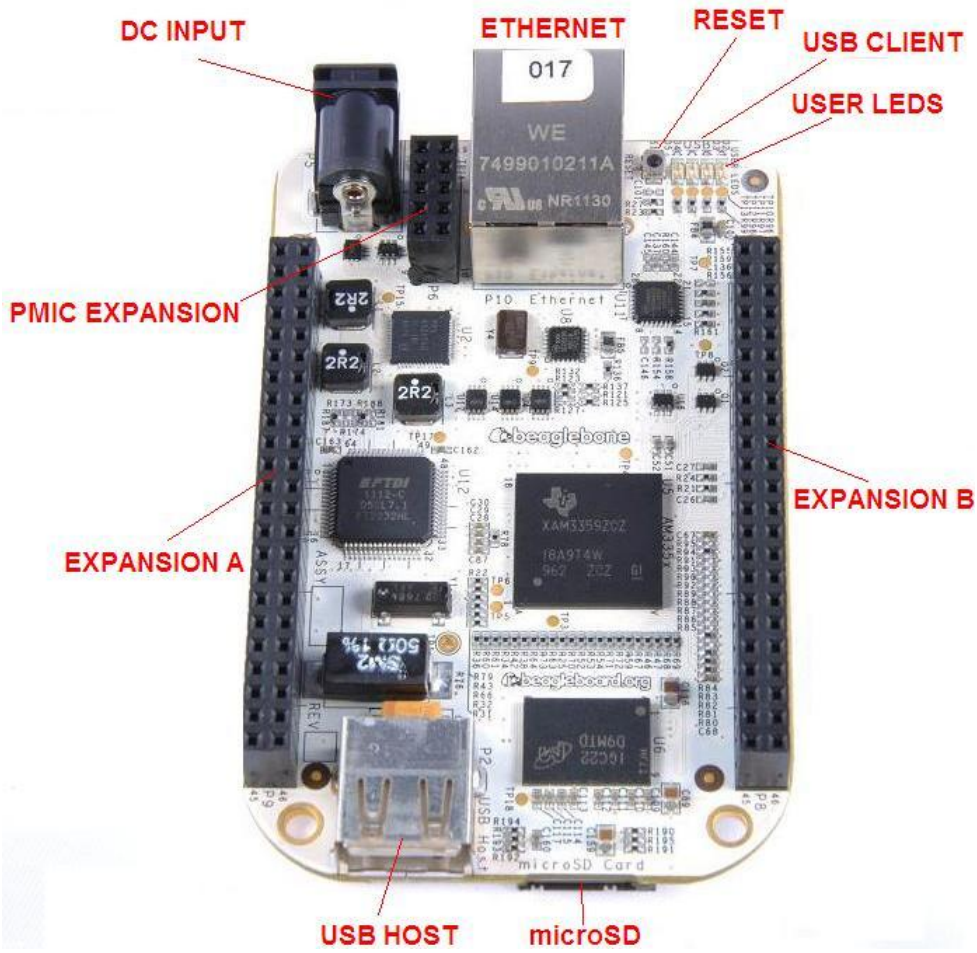


Figure 3. Board Connector and Indicators

NOTE: ***Be careful if you are considering using standoffs on the BeagleBone. The mounting hole next to the DC power jack has resistors that are a little too close to the hole and if you are not careful, you can damage those resistors when attaching the standoff. Use as small a diameter standoff as possible.***

4.0 BeagleBone Design Specification

This section provides a high level description of the design of the BeagleBone.

4.1 Processor

The board currently uses either the AM3359 or AM3358 processor in the 15x15 package. Actual processor speed will be determined by the actual devices supplied. The board is being released prior to the processor being in full production and as a result, has the AM3359 due to availability of those parts at this time. When changed to the AM3358, no loss of features will be experienced.

4.2 Memory

As single x16 bit DDR2 memory device is used. The design supports 128MB or 256MB of memory. The standard configuration is 256MB at 400MHz. A 128MB version may be built later, but there are no definite plans for this.

A single 32KB EEPROM is provided on I2C0 that holds the board information. This information includes board name, serial number, and revision information. Unused areas can be used by SW applications if desired.

4.3 Power Management

The **TPS65127B** power management device is used along with a separate LDO to provide power to the system.

4.4 PC USB Interface

The board will have an onboard USB HUB that concentrates two USB ports used on the board to one to facilitate the use of a single USB connector and cable to the PC. Support via this HUB includes:

- USB to serial debug
- USB to JTAG
- USB processor port access

When connected to the PC each of these will show up as ports on the PC.

4.4.1 Serial Debug Port

Serial debug is provided via UART0 on the processor using a dual channel FT2232H USB to serial device from FTDI to connect these signals to the USB port. Serial signals include Tx, Rx, RTS, and CTS.

A single EEPROM is provided on the FT2232H to allow for the programming of the vendor information so that when connected, the board can be identified and the appropriate driver installed.

4.4.2 JTAG Port

The second port on the FT2232H will be used for the JTAG port. Direct connection to the processor is made from the FT2232H. There is a JTAG header provided on the board as an option, but it is not populated.

4.4.3 USB0 Port

The HUB connects direct to the USB0 port on the processor. This allows that port to be accessible from the same USB connector as the Serial and JTAG ports.

4.5 MicroSD Connector

The board is equipped with a single microSD connector to act as the primary boot source for the board. A 4GB microSD card is supplied with each board. The connector will support larger capacity SD cards.

4.6 USB1 Port

On the board is a single USB Type A 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.

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

4.7 USB Client Port

Access to USB0 is provided via the onboard USB Hub. It will show up on a PC as a standard USB device.

4.8 Power Sources

The board can be powered from a USB port on a PC or from an optional 5VDC power supply. The power supply is not provided with the board and must be a grounded power supply. The USB cable is shipped with the board.

When powered from USB, the board is limited to 500 MHz. The onboard HUB + FT2232H power consumption does not leave room in the 500mA budget for the boot process. For 720 MHz operation, DC power is required. The lowest power mode is DC w/o the USB port connected, even at 720MHz.

Power can be supplied via a 2.1mm x 5.5mm center connector when connected to a positive power supply rated at 5VDC +/- .1V and 1A. This is similar to the power supply as currently used on BeagleBoards and the board can be powered from a supply that was used to power the BeagleBoard. Do not apply voltages in excess of 5V to the DC input. The DC power supply must be grounded.

4.9 Reset Button

When pressed and released, causes a reset of the board. Due to the small size of the switch, you will not experience a lot of travel when pushing the switch.

4.10 Indicators

There are five total green LEDs on the board. Four can be controlled by the user and one static LED.

- One power LED indicates that power is applied.
- Four Green LEDs that can be controlled via the SW by setting GPIO ports.

4.11 CTI JTAG Header

An optional 20 pin CTI JTAG header can be provided on the board to facilitate the SW development and debugging of the board by using various JTAG emulators. In order to use the connector, series resistors must be removed to isolate the USB to JTAG feature. This header is not supplied standard on the board and the typical user will not be able to make the resistor changes.

5.0 Expansion Interface

This section describes the expansion interface and the features and functions available from the expansion header.

5.1 Main Board Expansion Header

Two 46 pin dual row .1 x .1 female headers are supplied on the board for access to the expansion signals. Due to the number of pins, a low insertion force header has been chosen to facilitate the removal of the Capes. However, due to the large number of pins, removal can be difficult and care should be taken in the removal of the boards connected to the expansion headers. **Figure 4** below is a picture of the female header used.

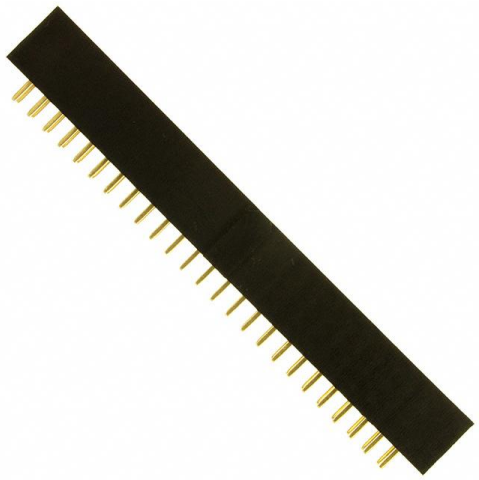


Figure 4. Main Board Expansion Connector

5.2 Cape Expansion Boards

Each expansion board or Cape will have 2 46 pin connectors. Their exact type and configuration will vary depending on the method used. Refer to **Section 8** for more details. The connectors used will be thruhole connectors.

Up to four Capes can be stacked onto the BeagleBone. Each board will have the same EEPROM as is found on the main board but will be at different addresses to allow for scanning for expansion boards via the I2C bus. Each board will be equipped with a 2 position dipswitch to set the address of the board based on the stack position. It is up to the user to insure the proper setting of this dipswitch to prevent a conflict on the I2C bus.

Standard expansion board size is 3.4" x 2.1". The board will have a notch in it to act as a key to insure proper orientation. The key is around the Ethernet connector on the main board.

Oversize boards, such as LCD panels, are allowed. The main board will extend out from under these boards.

5.3 Exposed Functions

This section covers functionality that is accessible from the expansion header.

NOTE: Not all functionality is available at the same time due to the extensive pin muxing of the signals on the processor.

Please refer to the processor documentation for detailed information on the uses and functions of the pins listed in the following sections.

5.3.1 LCD

A full 24 bit LCD panel can be supported. With the main board having backlight and touchscreen functionality, will simply and lower the cost of LCD expansion boards. Backlight power is limited to 25mA, so this may not be enough for larger panels.

If other functions are needed on an expansion board, such as NAND support, the full 24 bit display may not be able to be supported due to the pin muxing.

You can also create 16 bit LCD boards. The advantage here is that this uses fewer pins on the expansion connectors leaving more signals to be used by other expansion boards.

5.3.2 GPMC

Access to the GPMC bus is provided. Depending on the configuration needed, this may result in the loss of the LCD interface. Support for a 16 bit wide NAND is provided by the expansion board. This will limit the LCD display to 16Bits. Make sure you review and understand the pin muxing option before doing a design.

5.3.3 MMC1

MMC1 signals are exposed on the expansion headers.

5.3.4 SPI

There are two SPI ports available on the expansion header. SPIO0 has one CS and SPI1 has two CS signals.

5.3.5 I2C

There are two I2C Ports on the expansion header, I2C1 and I2C2. I2C2 is used for the EEPROMS on the expansion boards and must always be accessible. SW should never mess with these signals. Other components on a Cape can use this bus as long as it does not conflict with the base addresses of the Capes.

5.3.6 Serial Ports

There are four serial ports on the expansion headers. UART ports 1, 2, 4 ports have TX, Rx, RTS and CTS signals while UART5 only has TX and RX. UART 3 is NOT available for use.

5.3.7 A/D Converters

Seven 100K sample per second A to D converters are available on the expansion header.

NOTE: Maximum voltage is 1.8V. Do not exceed this voltage. Voltage dividers should be used for voltages higher than 1.8V.

In order to use these signals, level shifters will be required. These signals connect direct to the processor and care should be taken not to exceed this voltage.

The VDD_ADC voltage is 1.8V and is not to be used to power anything. It is only a reference voltage and should be used to set the reference level for those interfaces added to the CAPE and not used to supply power.

5.3.8 GPIO

A maximum of 66 GPIO pins are accessible from the expansion header. All of these pins are 3.3V and can be configured as inputs or outputs. Any GPIO can be used as an interrupt and is limited to two interrupts per GPIO Bank for a maximum of eight pins as interrupts.

5.3.9 CAN Bus

There are two can bus interfaces available on the expansion header supporting CAN version 2 parts A and B. The TX and RX digital signals are provided. The drivers and connectors will need to be provided on a daughter card for use.

5.3.10 TIMERS

There are four timer outputs on the expansion header.

5.3.11 PWM

There are up to eight PWM outputs on the expansion header.

- High Resolution Outputs- up to 6 single ended.
- ECAP PWM- 2 outputs

6.0 Detailed Board Design

This section describes the detailed design of the BeagleBone. Please be sure to reference the AM3359 datasheet and technical reference manual to gain a deeper understanding.

6.1 System Block Diagram

Figure 5 is the high level system block diagram of the BeagleBone.

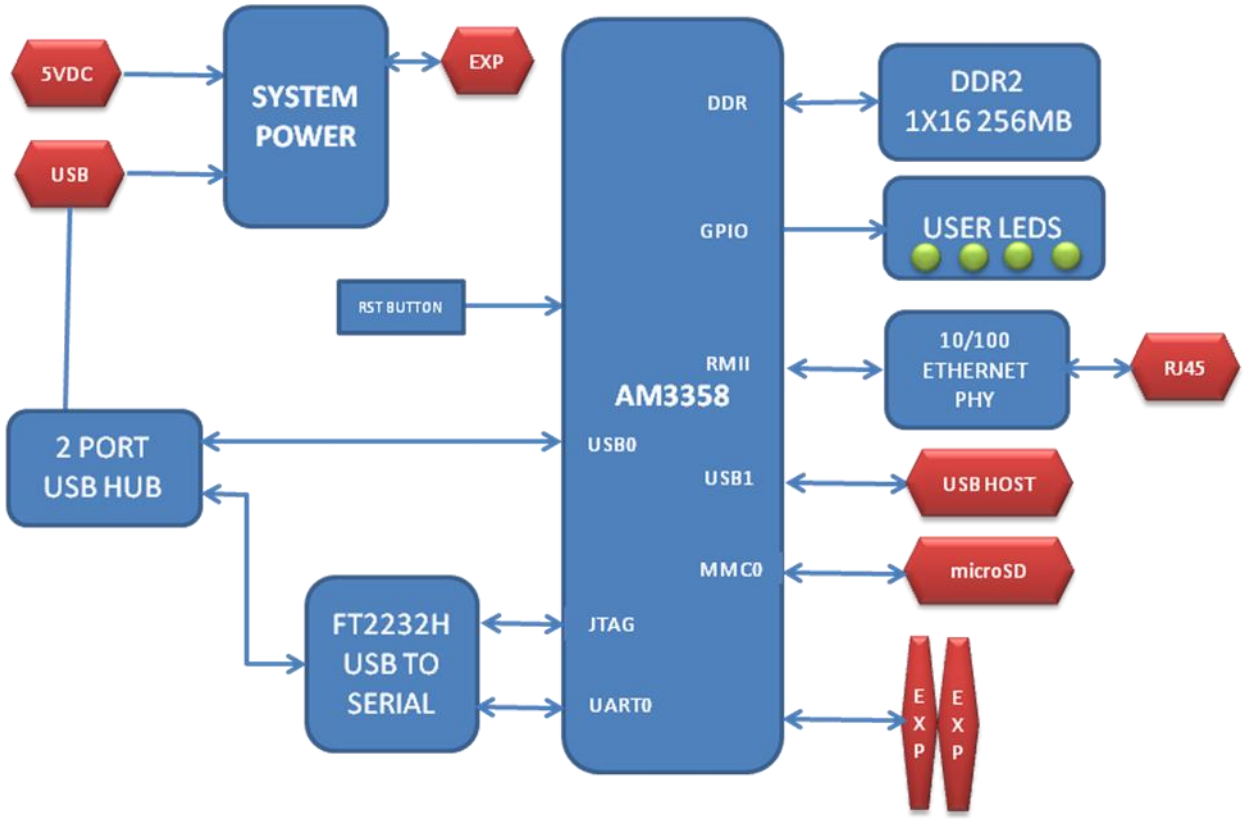


Figure 5. System Block Diagram

Each of these sections is discussed in more detail in the following sections.

6.2 Processor

The board is designed to use the AM3358 processor in the 15 x 15 package.

6.2.1 Processor Block Diagram

Figure 6 is a high level block diagram of the processor. For more information on the processor, go to http://www.ti.com/product/am3359

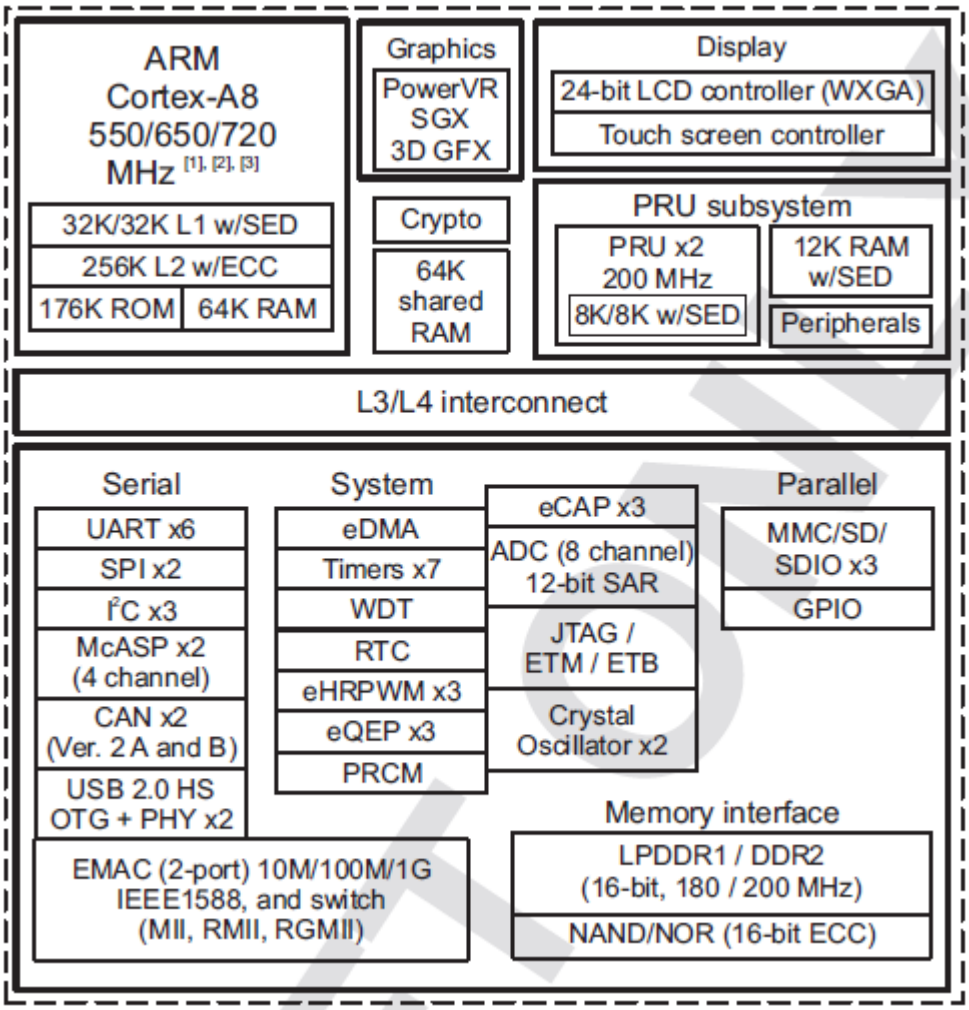


Figure 6. Processor Block Diagram

6.3 System Power

Figure 7 is a high level block diagram of the power section design of the BeagleBone.

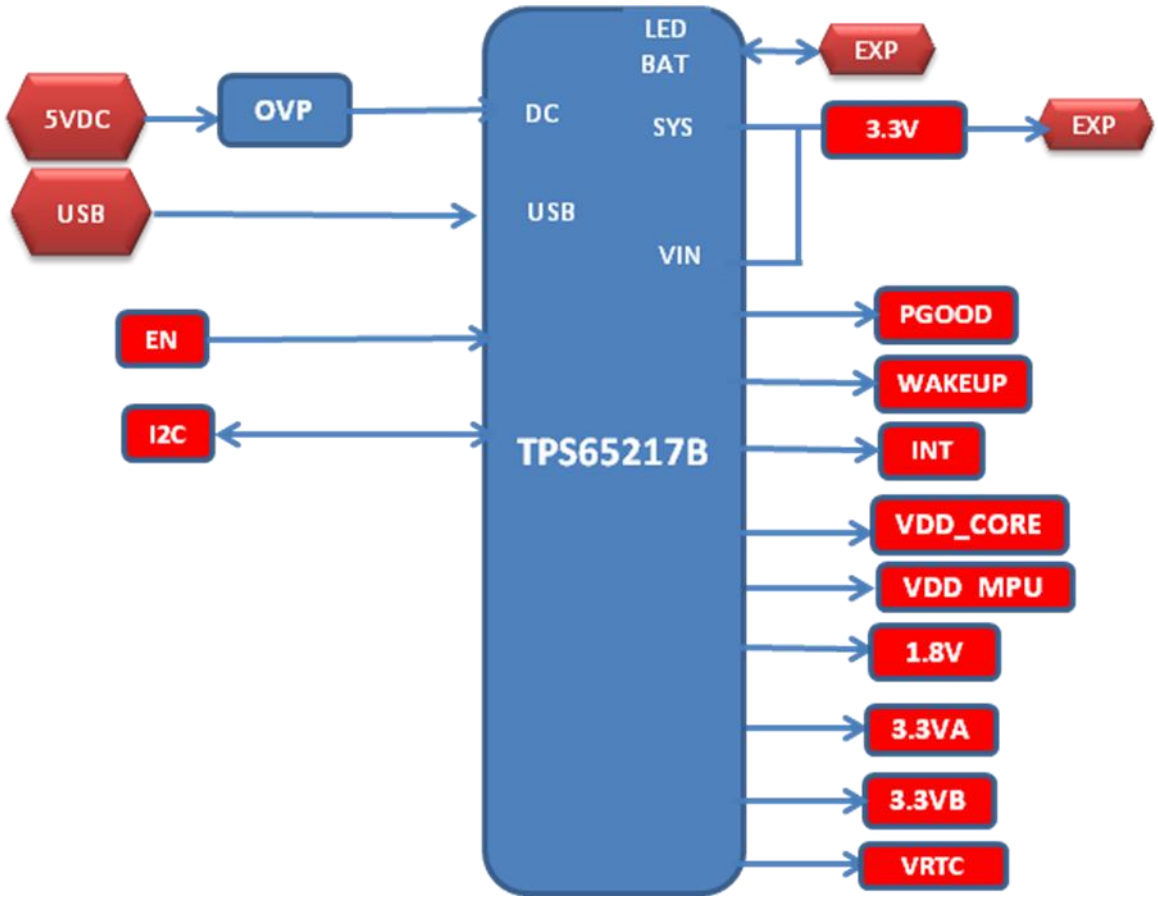


Figure 7. Power Subsection Block Diagram

6.3.1 TPS65217B PMIC

The main Power Management IC (PMIC) in the system is the TPS65217B. The TPS65217B is a single chip power management IC consisting of a linear dual-input power path, three step-down converters, four LDOs, and a high-efficiency boost converter to power two strings of up to 10 LEDs in series. 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 I_2C 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 SLEEP state output current is limited to 100uA to reduce quiescent current whereas 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 TPS65217B 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.

For more information on the TPS65217B, refer to http://www.ti.com/product/tps65217b . **Figure 8** is the high level block diagram of the TPS65217B.

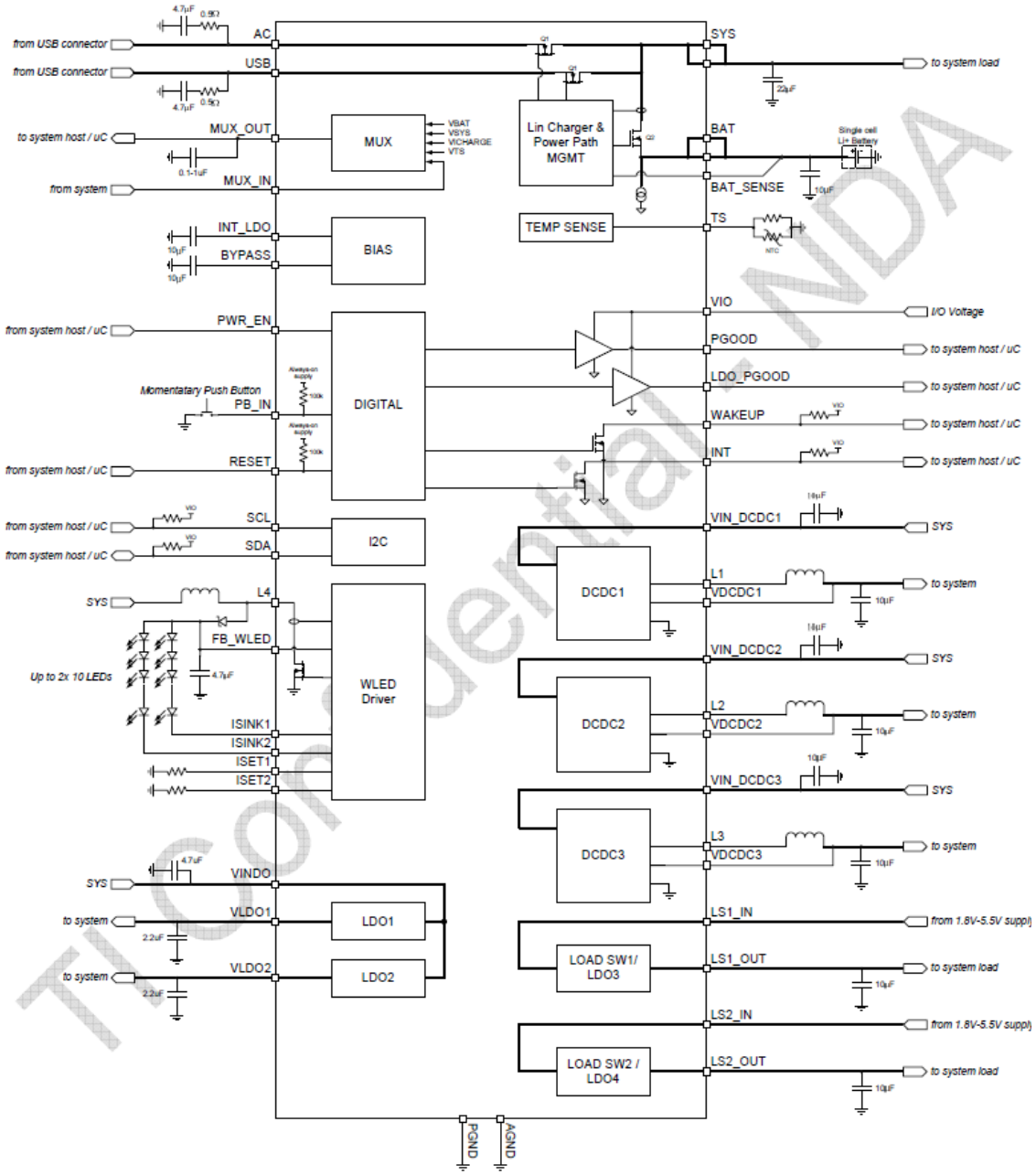


Figure 8. TPS65217B Block Diagram

6.3.2 5V DC Power Input

Figure 9 is the design of the 5V DC input circuit to the **TPS65217B**.

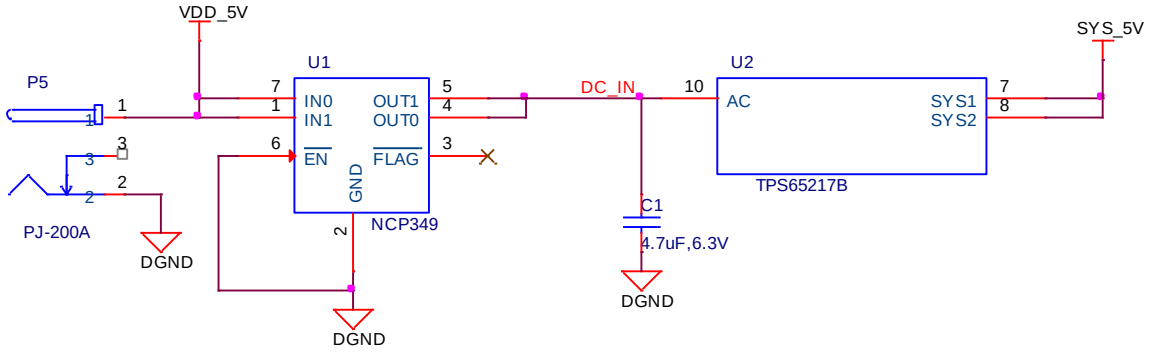


Figure 9. 5V DC Power Input

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. A **NCP349** over voltage device is used to prevent the plugging in of 7 to 12 V power supplies by mistake. The **NCP349** will shut down and the board will not power on. No visible indicator is provided to indicate that an over voltage condition exists. The board will not power up.

The 5VDC rail is connected to the expansion header. It is possible to power the board via the expansion headers from a 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.

6.3.3 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 that rail will not have that rail available for use. The 5VDC supply from the USB port is provided on the SYS_5V rail of the expansion header for use by a Cape. **Figure 10** is the design of the USB power input section.

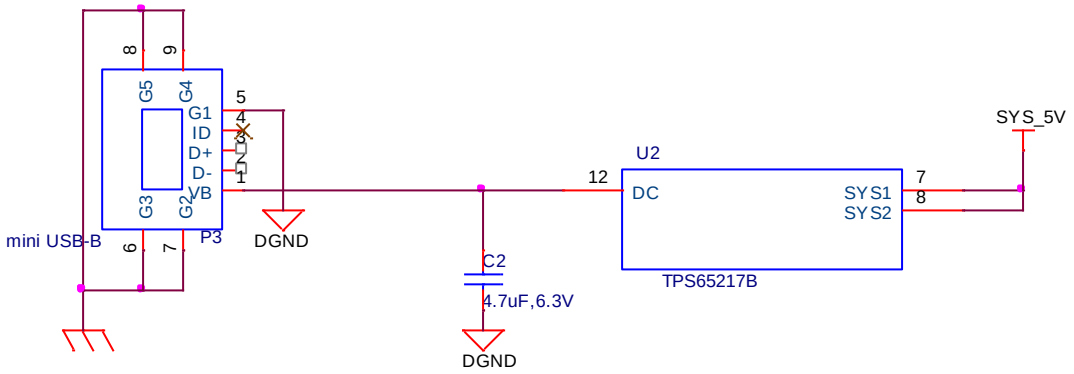


Figure 10. USB Power Input

6.3.4 Power Source Selection

The selection of either the 5VDC or the USB as the power source is handled internally to the **TPS65217B** 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 **TPS65217B** 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.

6.3.5 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 Power Consumption(mA@5V)

MODE	USB	DC	DC+USB
Reset	180	60	190
UBoot	363	230	340
Kernel Booting (Peak)	502	350	470
Kernel Idling	305	170	290

When the USB is connected, the FT2232 and HUB are powered up. This causes an increase in current. When the USB is not connected, these devices are in a lower power state. This is accounts for roughly 120mA of current and is the reason for the increased current when the USB is connected.

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

6.3.6 Power Sequencing

The power up process is made up of several stages and events. **Figure 11** is the events that make up the power up process of the system.

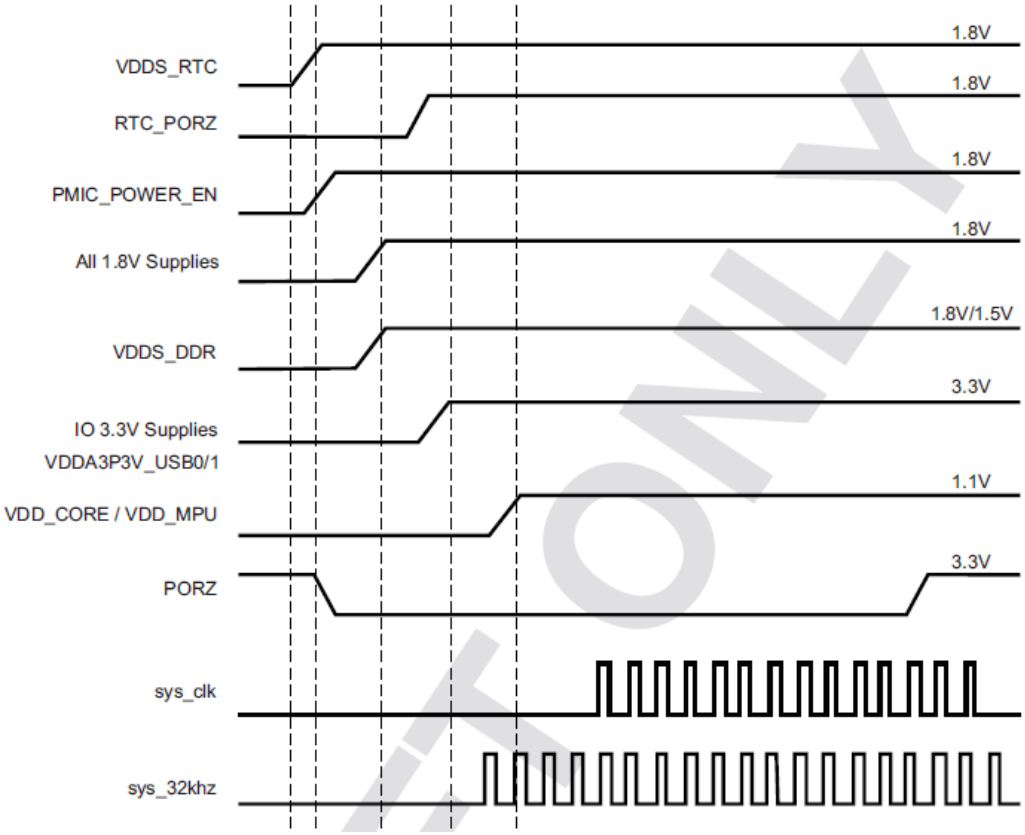


Figure 11. Power Sequencing

6.3.7 TPS65217B Power Up

When voltage is applied, DC or USB, the **TPS65217B** connects the power to the SYS output pin which drives the switchers and LDOS in the **TP65217B**.

At power up all switchers and LDOs are off except for the VRTC LDO (1.8V), provides power to the VRTC rail. Once the RTC rail powers up, the RTC_PORZ pin of the processor can be release. **Figure 12** is the circuit that controls the RTC_PORZ pin.

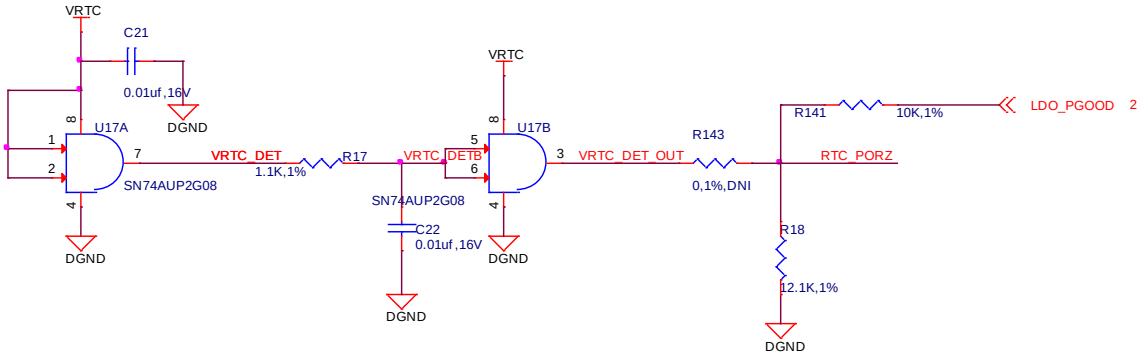


Figure 12. RTC_PORZ Control

There are actually two circuits in this design. One uses a pair of AND gates to create the **RTC_PORZ** signal and the other uses the **LDO_PGOOD** signal form the **TPS65217B**.

In the case of the AND gate circuit, once the VRTC rail comes up the circuit delays the **RTC_PORZ** which releases the RTC circuitry in the processor.

In the case of the **LDO_PGOOD** signal, it is provided by the **TPS65217B**. As this signal is 3.3V and the **RTC_PORZ** signal is 1.8V, a voltage divider is used. Once the LDOs are up on the TPS65217B, this signal goes active. The LDOs on the **TPS65217B** are used to power the VRTC rail on the processor.

The **LDO_PGOOD** version the default circuit currently used on the A3 design. It is possible on future revisions that the AND gate circuitry will be removed from the design.

Once the RTC block 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. This starts the **TPS65217B** power up process.

A separate signal, PMIC_PGOOD, holds the processor reset for 20ms after all power rails are up.

6.3.8 Voltage Rails

There are seven voltages supplied by the **TPS65217B**. Each of these are described in the following sections.

6.3.8.1 VDD_1V8

VDD_1V8 defaults to 1.8V on power up. The **TPS65217B** can deliver up to 1200mA on this rail. This rail only connects to the processor and the DDR2 memory.

6.3.8.2 VDD_MPU

VDD_MPU defaults to 1.1V on power up. This voltage can be changed under software control up to 1.25V in order to support higher processor frequencies. The **TPS65217B** can deliver up to 1200mA on this rail. This rail only connects to the processor.

6.3.8.3 VDD_CORE

VDD_CORE defaults to 1.1V on power up. This voltage should always be left at 1.1V. The **TPS65217B** can deliver up to 1200mA on this rail. This rail only connects to the processor.

6.3.8.4 VDD_3V3A

VDD_3V3A is the first of two 3.3V rails on the **TPS65217B**. The **TPS65217B** can deliver up to 225mA on this rail. This rail connects to the processor I/O rail voltage, TPS65217B I/O rail, and the SD/MMC card.

6.3.8.5 VDD_3V3B

VDD_3V3B is the second of two 3.3V rails on the **TPS65217B**. The TPS65217B can deliver up to 225 mA on this rail. This rail connects to the LAN8710, EEPROM, USB2412HUB, and FT2232.

6.3.8.6 VRTC

VRTC is the first rail to turn on during power up and is a 1.8V rail. The TPS65217B can deliver up to 100mA on this rail. This rail connects to the processor.

6.3.8.7 VLDO2

VLDO2 is a 3.3V rail that drives the power LED. This can be turned off via SW if a low current mode for the board, such as standby, is required.

6.3.9 Power Indicator LED

The board has a single power indicator LED. It is controlled via 3.3V VLDO2 power rail on the **TPS65217B**. When the **TPS65217B** has initialized and all switchers are on, the VLDO2 rail is activated turning on the LED. If the switchers are not initialized, for example if the processor does not enable the PWR_EN signal, the LED will not turn on. The power LED indicates that the **TPS65217B** is powered up. It is possible for the SW to turn off this rail to conserve power.

6.3.10 Expansion 3.3V LDO

A separate LDO provides the 3.3V rail to the expansion headers. **Figure 13** below is the design of the LDO.

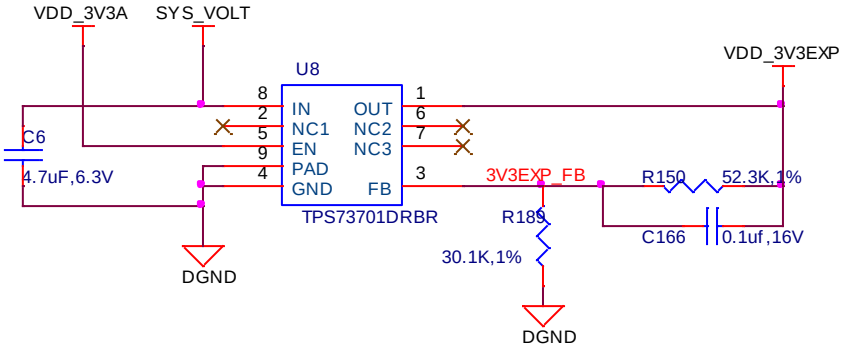


Figure 13. Expansion 3.3V Regulator

U8 is a **TPS73710** adjustable regulator that creates the 3.3V for the expansion bus by the values of **R150** and **R189**. The allowable current for this rail is set to **500mA** based on the design of the PCB, but that depends upon the total amount of current available from the main input supply. The LDO is cpapble of up to 1A of current.

6.4 Current Measurement

The BeagleBone has a method under which the current consumption of the board, not counting the USB Host port and expansion boards, can be measured. The voltage drop across a .1 ohm resistor is measured to determine the current consumption. **Figure 14** shows the interface to the **TPS65217B** to measure the current. The following sections describe this circuitry in more detail.

6.4.1 SYS_5V Connection

The **SYS_5V** rail is measured to determine the high side of the series resistor. The SYS_5V rail is connected to the MUX_OUT pin. Prior to being connected to the internal second multiplexer, the voltage is divided by 3. A 5V signal will result in a voltage of 1.66V at the MUX_OUT pin.

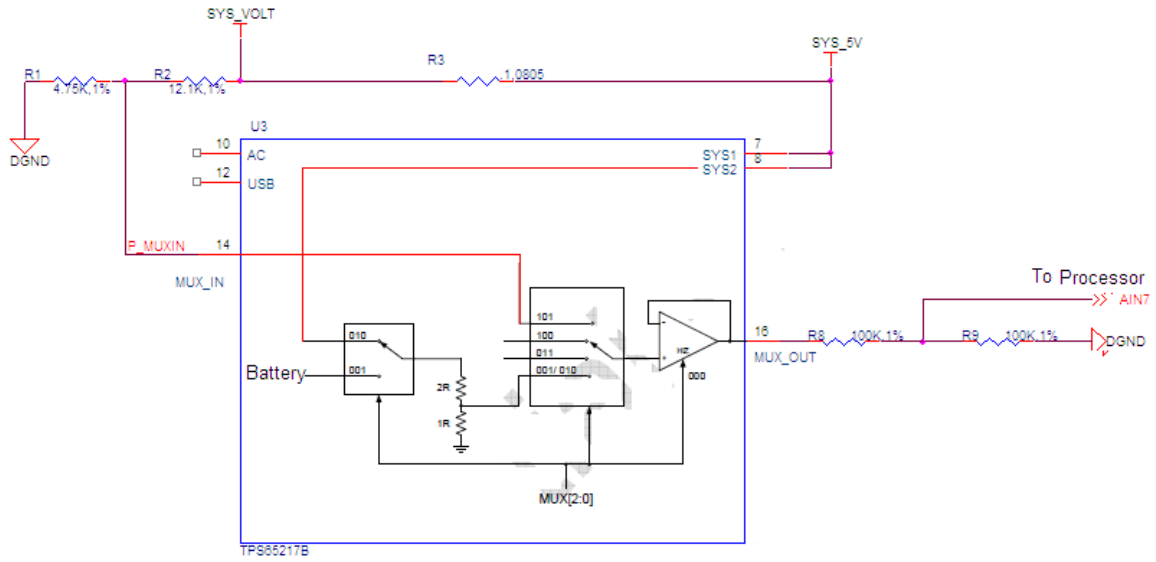


Figure 14. Current Measurement

6.4.2 SYS_VOLT Connection

The **SYS_VOLT** rail is measured to determine the high side of the series resistor. The **SYS_VOLT** rail is connected to the MUX_OUT by setting the registers inside the **TPS65217B**. The resistors **R2** and **R1** are provided to keep the same voltage divider configuration as found in the **SYS_5V** rail located internal to the **TPS65217B**. However, a 5V rail will give you 1.41V as opposed to the 1.66V found internal to the **TPS65217B**. This works out to a devisor of 2.8. Be sure and work this into your final calculations.

6.4.3 MUX_OUT Connection

The MUX_OUT connection is divided by 2 before being connected to the processor. The reason for this is that if the battery voltage is connected, it has no voltage divider internally. If connected it could damage the processor. When calculating the voltages for either side of the resistors, that voltage is divided by 2. Be sure and include this in your calculations.

6.4.4 Current Calculation

The calculation for the current is based on .1mV is equal to 1mA. You can use the following formula to calculate the current using the voltage readings as read by the processor.

(((SYS_5V*2)*3.3)-((SYS_VOLT*2)*3.54)))/.1=Total mA.

6.5 Two Port USB HUB

In order to provide access from a single USB port to the **FT2232** and the processor USB port, a SMSC **USB2412** dual port USB 2.0 HUB is provided. This device connects to the host PC.

Figure 15 is the design of the USB HUB.

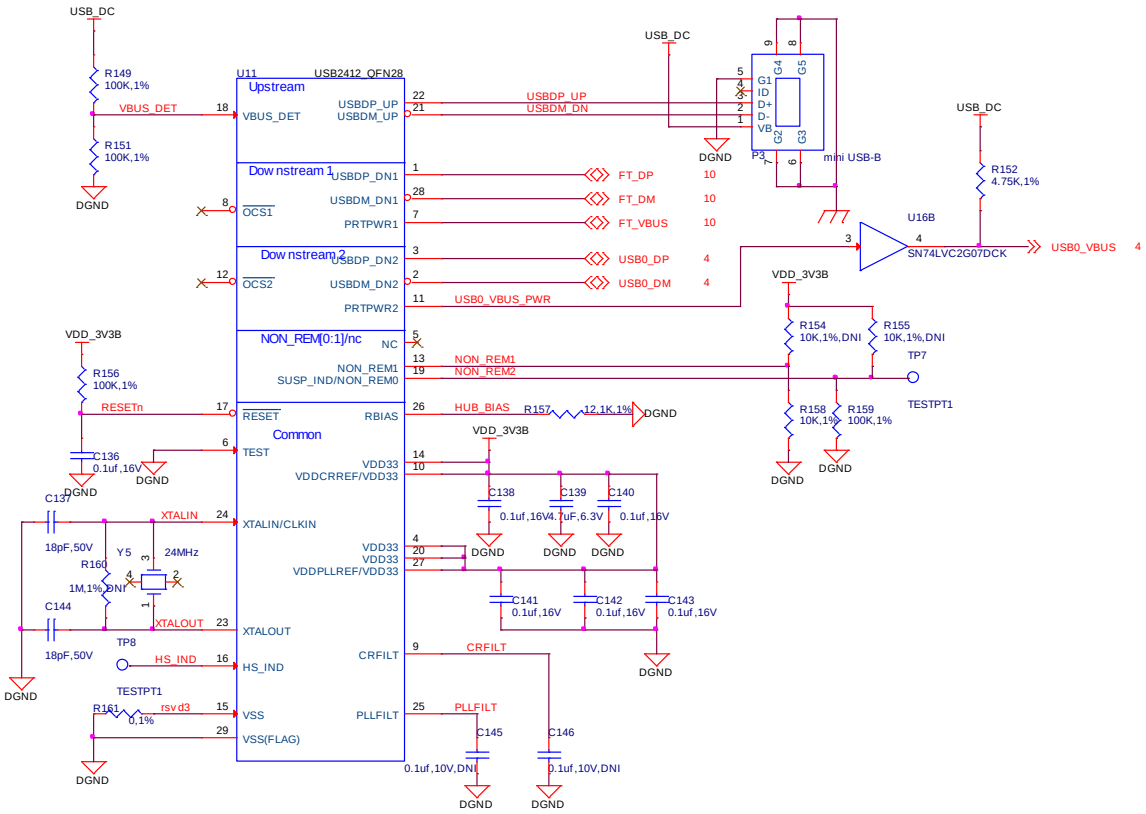


Figure 15. USB HUB Design

6.5.1 Processor USB Port

The USB connection to the host is via a mini USB connector. The power from this connector is connected to the **TPS65217B** to allow the board to be powered from the USB Host port. The signal pins connect to the USB HUB.

6.5.2 HUB Power

The HUB is powered from the 3.3VB rail from the **TPS65217B**. The HUB will remain in a low power mode until the USB port is connected. The **USB2412** monitors the **VBUS_DET** pin for logic high when the USB 5V supply is detected.

6.5.3 Crystal and Reset

The **USB2412** uses a single 24MHZ crystal. The RESET signal is self generated from the VDD_3V3B rail to an RC network.

6.5.4 FT2232H Serial Adapter

The first port of the HUB connected to the **FT2232** which handles the processor serial port and JTAG and is described in the next section. The DP and DM signals from the USB2412 connect direct to the **FT2232H**. The FT_BUS signal is used by the **FT2232H** to detect the presence of the host USB port. Once the HUB is connected to the Host, this pin will go HI to indicate the presence of the USB port.

6.5.5 Processor USB Port

The second port of the HUB is connected to the processor USB port 0. In order for the port to work on the processor it must first detect the presence of 5V on the **VBUS** pin. The USB2412 puts out a 3.3V signal on the **PRTPWR2** so **U16** converts that signal to a 5V logic level as required by the processor.

6.6 FT2232H USB to Serial Adapter

The **FT2232H** from FTDI provides the conversion from the USB port to the JTAG interface and Serial port to the processor. **Figure 16** is the design of the FT2232H circuit.

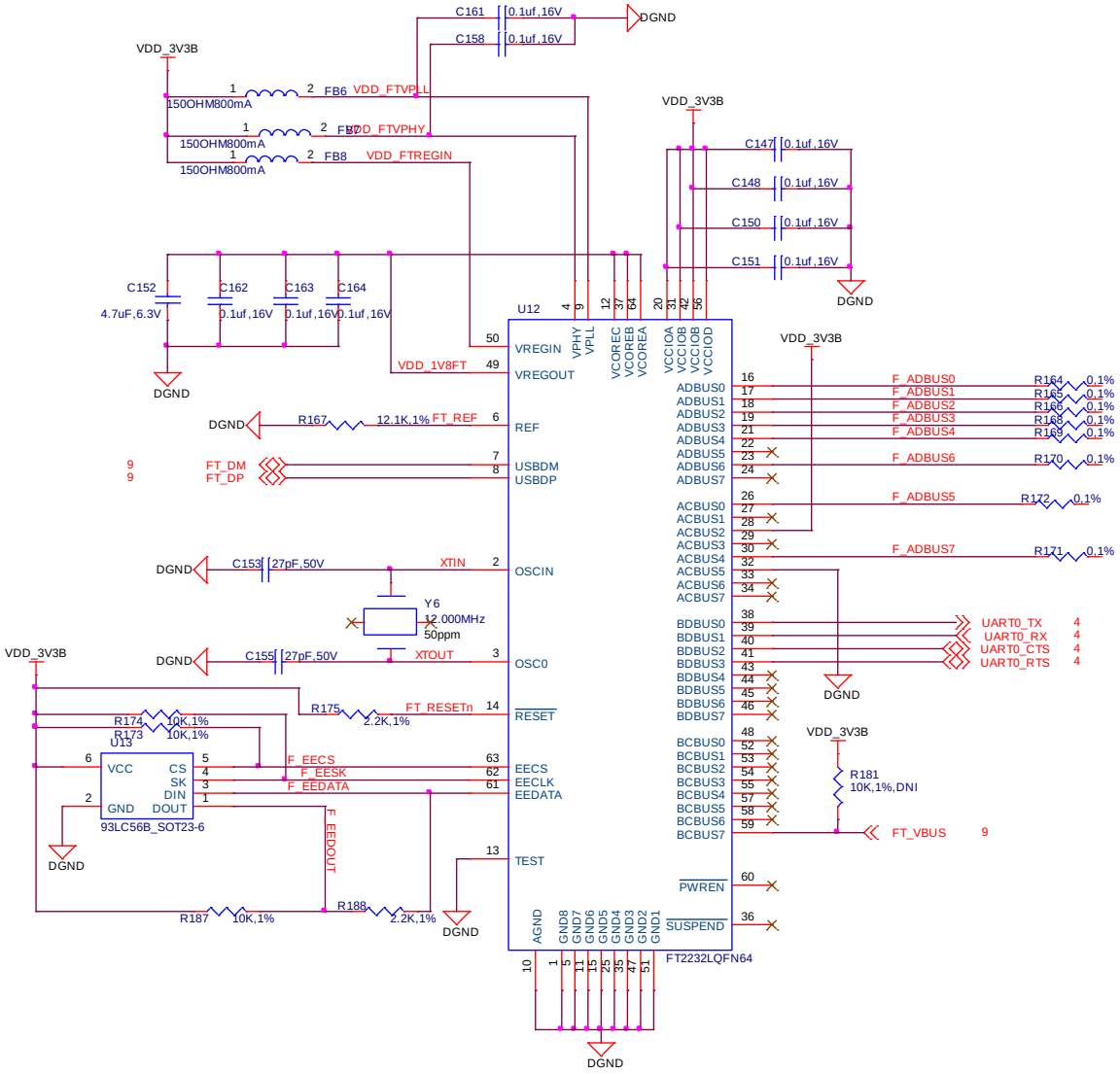


Figure 16. FT2232H Design

6.6.1 EEPROM

U13 is a EEPROM that tells **U12** the configuration of the device and the I/O pins. In order for the FT2232H to operate properly, this device must be programmed. Using the tools provided by FTDI makes this process straight forward.

6.6.2 JTAG

Using a parallel I/O mode, the **FT2232H** can be used to access the **JTAG** signals on the processor. At USB 2.0 speeds, the throughput is very good, and should provide connectivity to several popular debug environments including Code Composer Studio.

On the **Rev A6** the reset from the FT2232 has been disabled. This is due to spurious reset signals being generated by the FT2232 on target connect when using Code Composer Studio.

6.6.3 Serial Port

Access to **UART0** is provided by the **FT2232H** via the USB port. Signals available are TX, RX, RTS, and CTS.

6.7 256MB DDR2 Memory

The board comes standard with 256MB DDR SDRAM configured as a single 128M x 16 device. The design will also support a single 64M x 16 device for 128MB of memory. The memory size cannot be extended past 256MB. The design uses a single MT47H128M16RT-25E:C 400MHZ memory from Micron which comes in an 84-Ball 9.0mm x 12.5mm FBGA package. **Table 4** below is the addressing configuration of the device.

Table 4. DDR Addressing

Parameter	128 Meg x 16
Configuration	16 Meg x 16 x 8 banks
Refresh count	8K
Row address	A[13:0] (16K)
Bank address	BA[2:0] (8)
Column address	A[9:0] (1K)

Figure 17 is the functional block diagram of the DDR2 memory device.

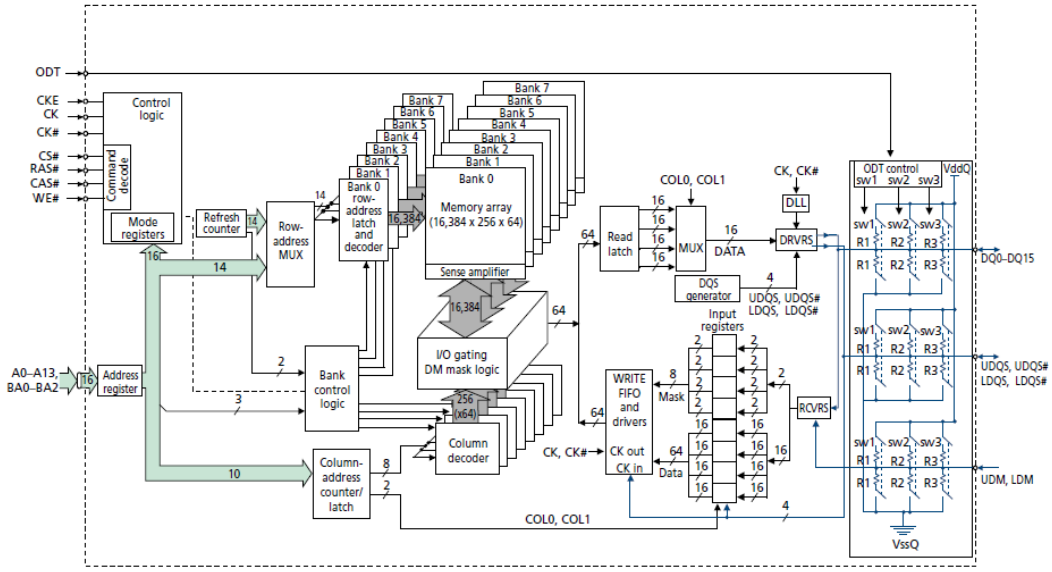


Figure 17. DDR Device Block Diagram

6.7.1 DDR 2 Design

Figure 18 below is the schematic of the DDR implementation. The memory is placed as close to the processor as possible to minimize layout and signal issues.

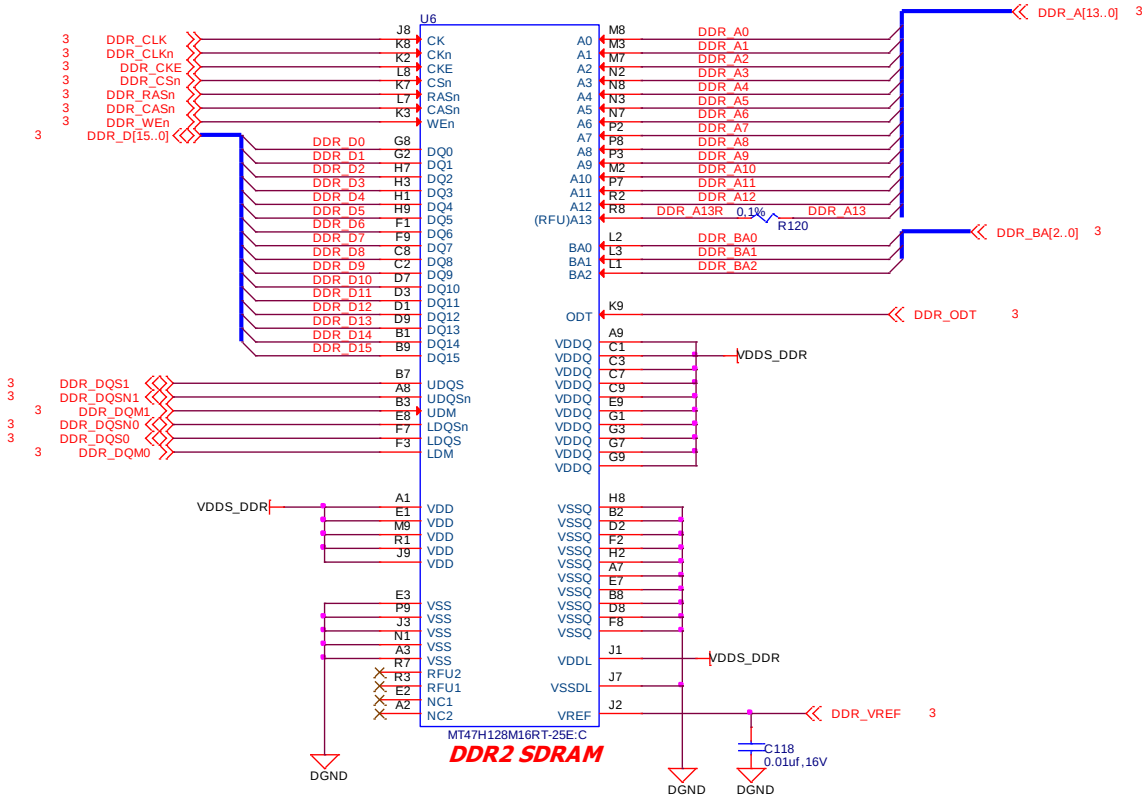


Figure 18. DDR Design

The DDR2 connects direct to the processor and no external interface devices are required. Power is supplied to the **DDR2** via the 1.8V rail on the **TPS65217B**.

6.7.2 DDR VTP Termination Resistor

There is a requirement for a 50 ohm 1% termination resistor, **R76**, on the DDR interface. You will notice that the one used on the board design is a 50W wire wound resistor. The reason for this is cost. This resistor can be expensive and at the time of the design, this was the least expensive one package available. On the Rev A4 design, we added two more resistors, R217 and R218, to allow for a 0603 and 0805 package for applications where space is critical and to give us more options where parts availability is concerned.

6.7.3 User LEDs

Four user LEDS are provided via GPIO pins on the processor. **Figure 19** below shows the LED circuitry.

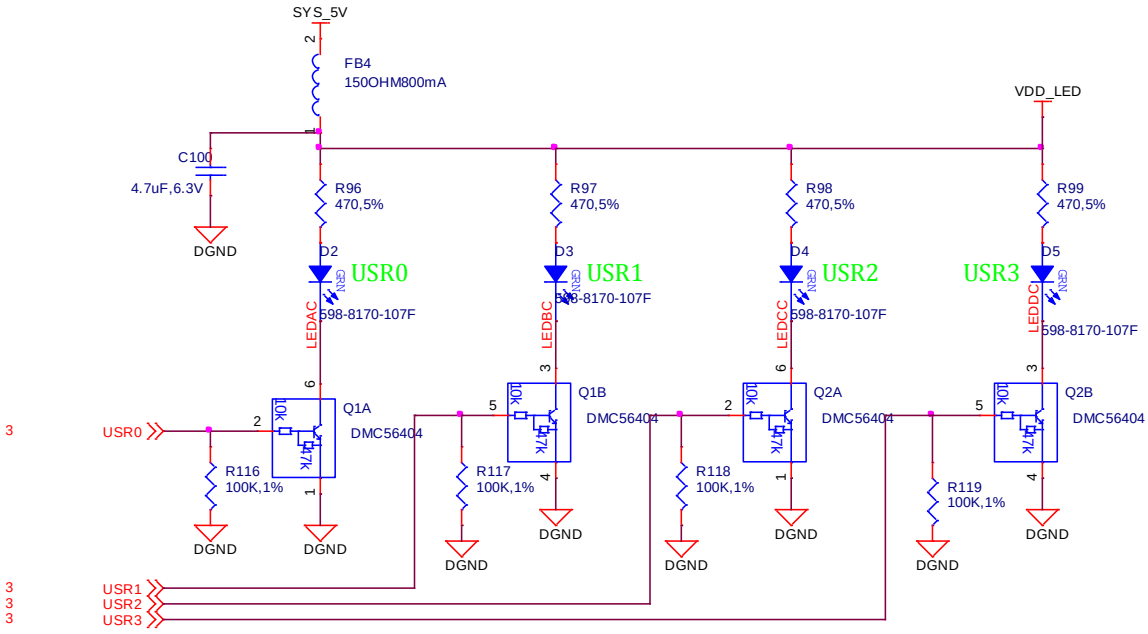


Figure 19. User LEDS

Q1 and **Q2** provide level shifting from the processor to drive the LEDs that are connected the **SYS_5V** rail. **FB4** provides noise immunity to the system by the LEDS which can be a source of noise back into the system rail. Each LED is controlled by setting the appropriate GPIO bit HI. At power up all LEDs are off. **Table 5** is the GPIO USER LED assignments.

Table 5. User LED Control

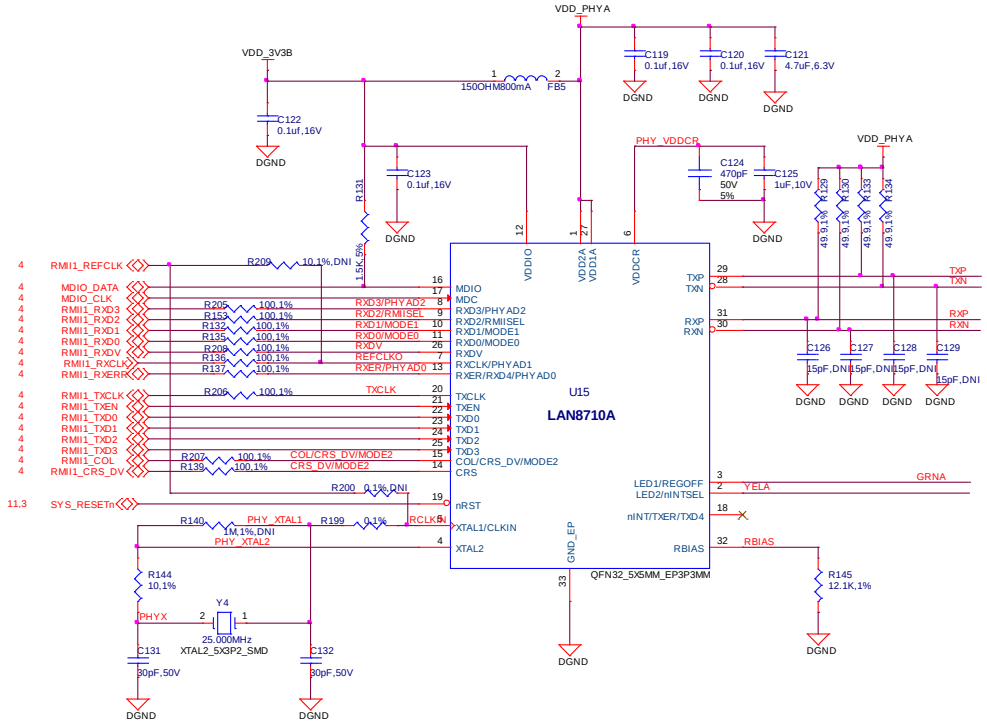
LED	GPIO
User 0	GPIO1_21
User 1	GPIO1_22
User 2	GPIO1_23
User 3	GPIO1_24

6.8 10/100 Ethernet

The 10/100 Ethernet uses a SMSC LAN8710A Ethernet PHY and interfaces to the processor using the MII interface. This section covers that design.

6.8.1 Ethernet PHY Design

Figure 20 below is the design of the 10/100 PHY section of the board.



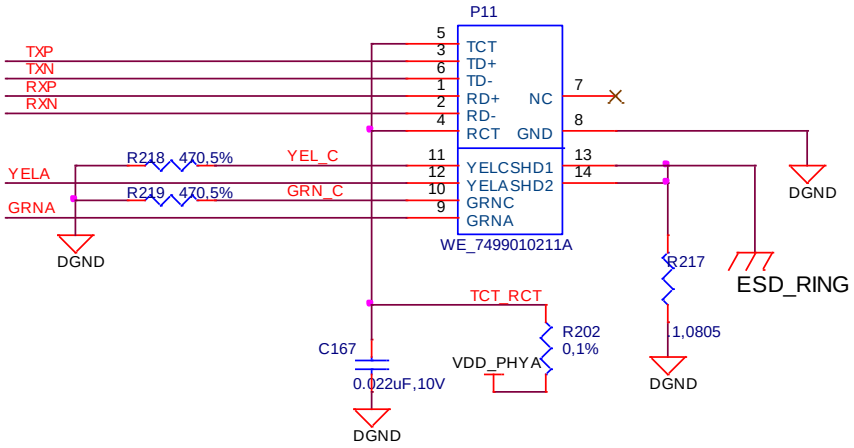


Figure 20. 10/100 Ethernet PHY Design

6.8.2 Processor Signal Description

The **Table 6** describes the signals between the processor and the LAN8710A. The BALL column is the pin number on the processor. The SIGNAL name is the generic name of the signal on the processor. The PHY column is the pin number of the PHY.

Table 6. Processor Ethernet Signals

SIGNAL NAME	DESCRIPTION	TYPE	BALL	PHY
gmii1_col	MII Collision	I	H16	15
gmii1_crs	MII Carrier Sense	I	H17	14
gmii1_rxclk	MII Receive Clock	I	L18	7
gmii1_rxd0	MII Receive Data bit 0	I	M16	11
gmii1_rxd1	MII Receive Data bit 1	I	L15	10
gmii1_rxd2	MII Receive Data bit 2	I	L16	9
gmii1_rxd3	MII Receive Data bit 3	I	L17	8
gmii1_rxdv	MII Receive Data Valid	I	J17	26
gmii1_rxer	MII Receive Data Error	I	J15	13
gmii1_txclk	MII Transmit Clock	I	K18	20
gmii1_txd0	MII Transmit Data bit 0	O	K17	22
gmii1_txd1	MII Transmit Data bit 1	O	K16	23
gmii1_txd2	MII Transmit Data bit 2	O	K15	24
gmii1_txd3	MII Transmit Data bit 3	O	J18	25
gmii1_txen	MII Transmit Enable	O	J16	21
MDC	MDIO Clock	O	M18	17
MDIO	MDIO Data	I/O	M17	16

6.8.3 Clocking Mode

The LAN8710A provides the clock to the processor and is generated by the onboard 25MHz crystal **Y4**. There are independent clocks for the transmit channel (**MII Transmit Clock**) and for the receive channel (**MII Receive clock**).

6.8.4 PHY Mode

The PHY operates in the 10/100 mode with auto negotiation enabled. This is set via the resistors as described in **Figure 21** which are sampled by the PHY when coming out of reset. It is possible for SW to override this setting if required by setting these bits via the MDIO channel.

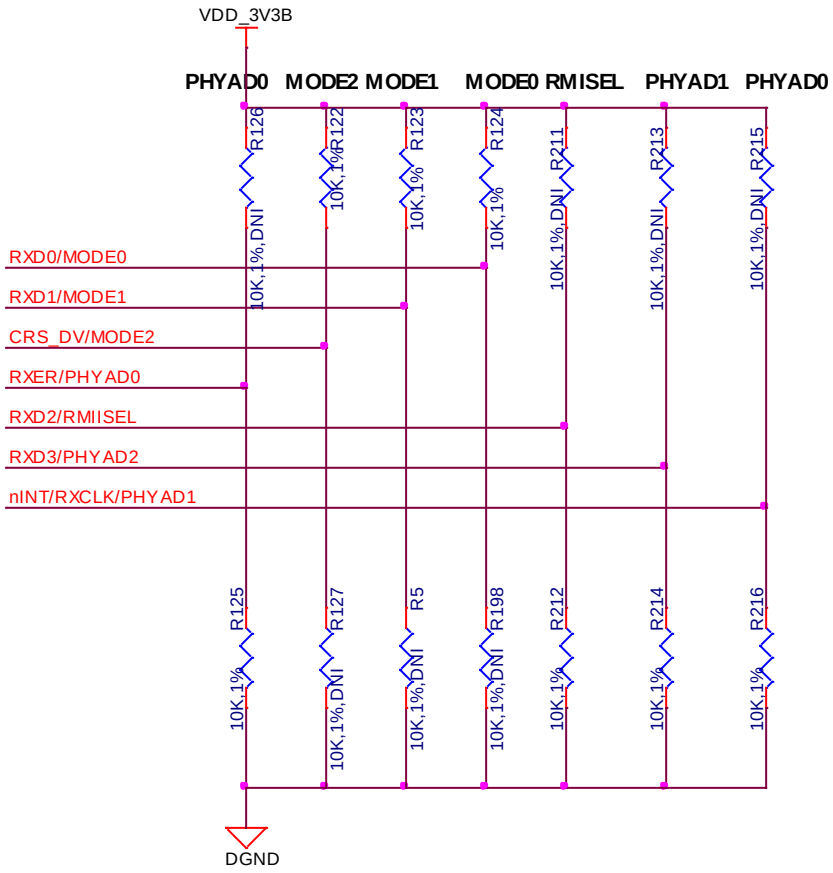


Figure 21. 10/100 Ethernet PHY Default Settings

By adding pull up or pull down resistors, the default mode of the PHY can be set via HW. Seven pairs of resistors are provided on the board to set the mode.

Pins **MODE0-1** set the operating mode of the PHY. Default mode is intended to be set by the populating of **R122-124** to 111 enabling all operating modes and auto negotiation. However, there is an error on that the CRS_DV/MODE2 signal is connected to pin 14 on

the SMSC PHY. This is incorrect. It should be connected to pin 15. As a result the mode is 011 which sets it to 100M and no auto negotiate. The SW should overwrite the register in initial setup, so this should not cause any operational issues. This will be changed on the next revision of the board assuming something else is required for a change. This issue is not deemed sufficient to warrant another revision of the board at this time. No operational issues have been identified as a result of this error.

PHYAD0-2 sets the default address of the PHY. Populating **R124-R126** set the default of 0. It is not expected to be set to anything other than this, but the other option was enabled just in case.

RMIISEL sets the mode to RMII if **R211** is installed. MII is the default mode used in this design, so **R212** needs to be installed and **R211** is not to be installed.

6.8.5 MDIO Interface

The MDIO interface is the control channel interface between the processor and the **LAN8710A**. Via this interface all of the internal PHY registers can be read and set by the processor and important status information can be read.

6.8.6 PHY Reset

The PHY reset signal is connected to the main board reset and is reset on power up.

6.8.7 Status LEDs

They Ethernet connector has a Yellow and Green LED. The Green LED will be on when a link is established. It flashes off when data is transferred.

The Yellow status LED will work differently for each revision.

A3...The Yellow LED is OFF when the link is 100M and ON when it is 10M.

A4...The Yellow LED is ON when the link is 100M and OFF when it is 10M. However, after removing R219 which is required, operation reverts back to the same as A3.

A5… Yellow LED is OFF when the link is 100M and ON when it is 10M.

A6….. The Yellow LED is ON when the link is 100M and OFF when it is 10M

6.8.8 Power

The PHY is powered via the 3.3VB rail from the **TPS65217B**. A filter is provided between the 3.3VB rail and the PHY. The internal LDO is used to power the internal rails.

6.9 USB Host

A single USB Host port is provided on the board. It is driven by USB port 1 of the processor. The port can deliver up to 500mA of current provided that much current is available from the power supply. In the scenario where the board is totally powered from the USB input, the power supplied will be much less and dependent on how much current is available after driving the board and any daughter cards that may be attached.

6.9.1 USB Host design

The board has a single USB host connector accessible via **P2** a type A female connector. **Figure 22** below is the USB Host design.

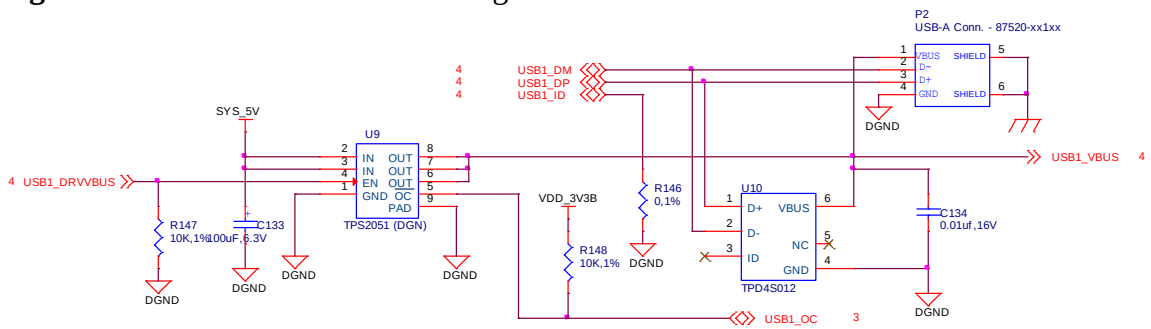


Figure 22. USB Host Design

The USB port on the processor is an OTG port. In order to force the host function needed, the ID pin, **USB_ID**, is grounded permanently by **R146**.

U9, a **TPS2051,** is the power switch that controls the 5VDC to the USB port. It is turned on by the processor via **USB1_DRVVBUS** signal. The USB1_VBUS signal is a confirmation back to the processor that the switch is activated and that 5V is connected to the USB Host connector.

In the event of an over current condition, the switch will signal the processor of the event, via **USB1_OC**, and the switch will shut down. **R148** is a pullup to provide the HI voltage level because the **OC** signal on **U9** is an open drain pin. **C133** provides extra current when devices are inserted into the connector per the USB specification. The amount of current the switch can provide is limited by the available current from the main power source. In order to handle high current devices, you need to power the board from the DC input connector and not USB. Powering from USB can in, most cases, supply enough current to run a thumbdrive or low current device.

U10 is an ESD protection device intended to protect the processor.

6.10 SD Connector

The board is populated with a microSD small form factor SD slot. It will support High capacity cards. The voltage rail for the connector is **3.3VA**. A card detector output is provided from the connector to the CD/EMU4 signal. **Figure 23** shows the connections to the microSD connector.

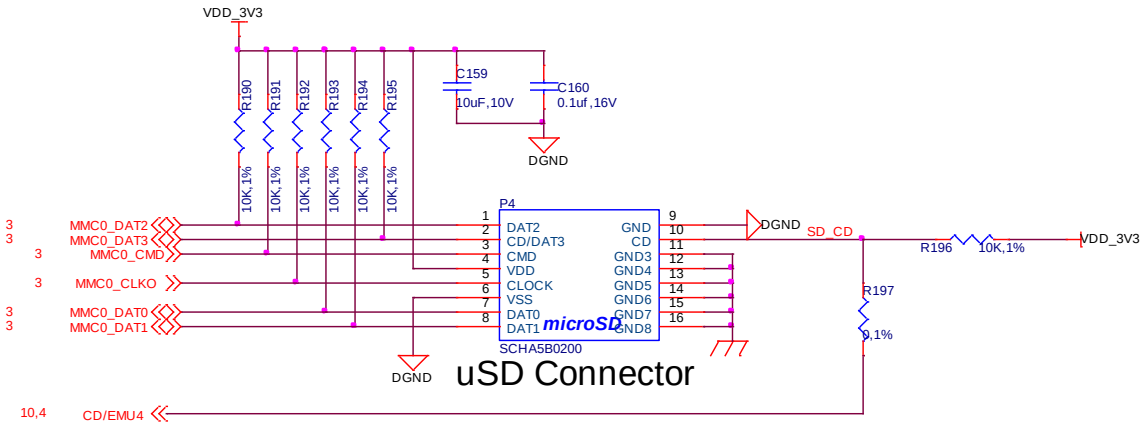


Figure 23. SD Connector Design

There are pullup resistors on all the signals to provide additional drive strength and to increase the rise time of the signals. The **SD_CD** is the signal that indicates to the processor that the card is inserted. The signal is a contact point on the connector and **R196** provides the logic hi signal that is grounded whenever there is no card inserted. When the card is inserted, the signal will go high. **R197** is provided as an option to allow this signal to be removed from the processor for use as the **EMU**4 signal by the optional JTAG connector.

The connector is located on the bottom side of the board and the card should be inserted with the label side up and the contact pins down. This connector is a Push-Push connector. To insert the card push the card in until it clicks and then release. To remove the card, push the card in and the connector will release the card and eject the card.

WARNING: DO NOT PULL THE CARD OUT TO REMOVE IT OR YOU MAY DAMAGE THE CONNECTOR.

6.11 EEPROM

The BeagleBone is equipped with a single CAT24C256W EEPROM to allow the SW to identify the board. **Table 7** below defined the contents of the EERPOM.

Table 7. EEPROM Contents

Name	Size (bytes)	Contents
Header	4	0xAA, 0x55, 0x33, EE
Board Name	8	Name for board in ASCII: A335BONE
Version	4	Hardware version code for board in ASCII: A3 for Rev A3, 00A4 for Rev A4, 00A5 for Rev A5, 00A6 for Rev A6.
Serial Number	12	Serial number of the board. This is a 12 character string which is: WWYY4P16nnnn where: WW = 2 digit week of the year of production YY = 2 digit year of production nnnn = incrementing board number
Configuration Option	32	Codes to show the configuration setup on this board. 00000000000000000000000000000000
RSVD	6	000000
RSVD	6	000000
RSVD	6	000000
Available	32702	Available space for other non-volatile codes/data

Figure 24 is the design of the EEPROM circuit as it is found on the Rev A3, A4, and A5 versions.

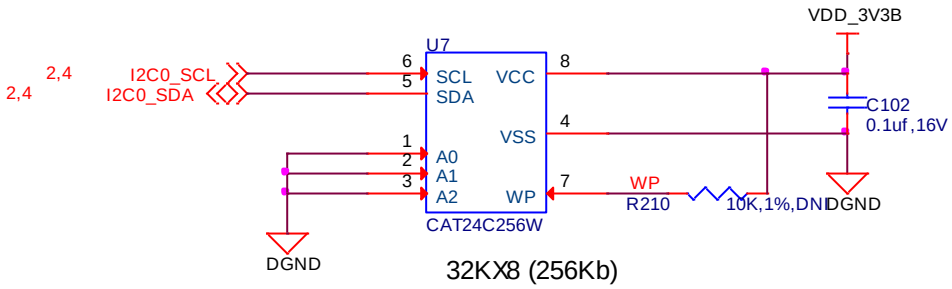


Figure 24. EEPROM Design Rev A3,A4, and A5

Figure 25 shows the new design on the **Rev A6** where the **WP** is implemented and a test point is provided to bypass it.

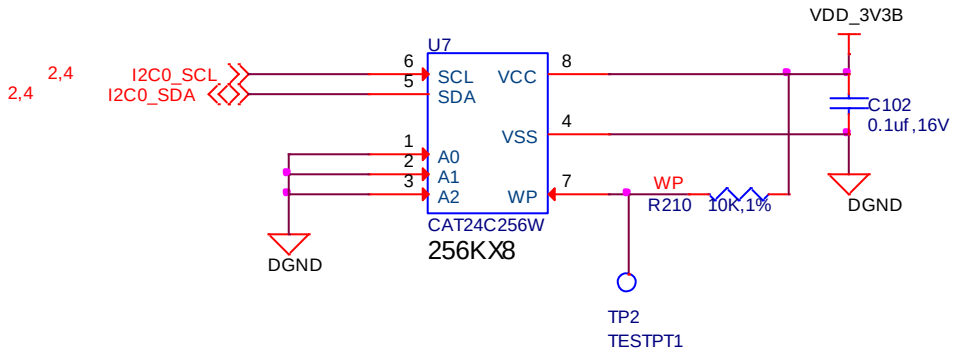


Figure 25. EEPROM Design Rev A6

The EEPROM is accessed by the processor using the I2C 0 bus. The EEPROM is not write protected on Revision A3, A4, and A5. **R210** is installed on Revision A6 which invokes the write protection. The **WP** pin has an internal pulldown on it so that if removed the pin is grounded. By grounding the test point, the write protection is removed/

The first 48 locations should not be written to if you choose to use the extras storage space in the EEPROM for other purposes. If you do, it could prevent the board from booting properly as the SW uses this information to determine how to set up the board.

6.12 ADC Interface

The processor has 8 ADC (Analog to Digital) converter inputs. The signals are 1.8V only interfaces. One of these, AD7, is connected to the **TPS65217B** and used for measuring voltages and current via the **TPS65217B**.

6.12.1 ADC Inputs

The primary purpose of the ADC pins was intended for use as a Touchscreen controller but can be used as a general purpose ADC. Each signal is a 12b successive approximation register (SAR) ADC. Sample rate is 100K samples per second. There is only one ADC in the processor and it can be connected to any of the 8 ADC pins.

6.12.2 VDD_ADC Interface

The signal VDD_ADC is provided via the expansion header, but is not a voltage rail that is to be used to power anything on an expansion board. It is supplied from the 1.8V rail of the **TPS65217B** and is run through an inductor for noise isolation. It is there if need for external circuitry to have access to the VREF rail of the ADC or to add additional filtering via a capacitor if needed.

6.13 Expansion Headers

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.

6.13.1 Expansion Header P8

Table 8 shows the default 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.

Table 8. Expansion Header P8 Pinout

SIGNAL NAME	PROC	CONN		PROC	SIGNAL NAME
	GND	1	2	GND	
GPIO1_6	R9	3	4	T9	GPIO1_7
GPIO1_2	R8	5	6	T8	GPIO1_3
TIMER4	R7	7	8	T7	TIMER7
TIMER5	T6	9	10	U6	TIMER6
GPIO1_13	R12	11	12	T12	GPIO1_12
EHRPWM2B	T10	13	14	T11	GPIO0_26
GPIO1_15	U13	15	16	V13	GPIO1_14
GPIO0_27	U12	17	18	V12	GPIO2_1
EHRPWM2A	U10	19	20	V9	GPIO1_31
GPIO1_30	U9	21	22	V8	GPIO1_5
GPIO1_4	U8	23	24	V7	GPIO1_1
GPIO1_0	U7	25	26	V6	GPIO1_29
GPIO2_22	U5	27	28	V5	GPIO2_24
GPIO2_23	R5	29	30	R6	GPIO2_25
UART5_CTSN	V4	31	32	T5	UART5_RTSN
UART4_RTSN	V3	33	34	U4	UART3_RTSN
UART4_CTSN	V2	35	36	U3	UART3_CTSN
UART5_TXD	U1	37	38	U2	UART5_RXD
GPIO2_12	T3	39	40	T4	GPIO2_13
GPIO2_10	T1	41	42	T2	GPIO2_11
GPIO2_8	R3	43	44	R4	GPIO2_9
GPIO2_6	R1	45	46	R2	GPIO2_7

6.13.2 P8 Signal Pin Mux Options

Table 9 shows the other signals that can be connected to each pin of **P8** based on the settings of the registers in the processor for modes 0-3.

Table 9. P8 Mux Options Modes 0-3

PIN	PROC	NAME	MODE0	MODE1	MODE2	MODE3
1		GND				
2		GND				
3	R9	GPIO1_6	gpmc_ad6	mmc1_dat6		
4	T9	GPIO1_7	gpmc_ad7	mmc1_dat7		
5	R8	GPIO1_2	gpmc_ad2	mmc1_dat2		
6	T8	GPIO1_3	gpmc_ad3	mmc1_dat3		
7	R7	TIMER4	gpmc_advn_ale		timer4	
8	T7	TIMER7	gpmc_oen_ren		timer7	
9	T6	TIMER5	gpmc_be0n_cle		timer5	
10	U6	TIMER6	gpmc_wen		timer6	
11	R12	GPIO1_13	gpmc_ad13	lcd_data18	mmc1_dat5	mmc2_dat1
12	T12	GPIO1_12	GPMC_AD12	LCD_DATA19	MMC1_DAT4	MMC2_DAT0
13	T10	EHRPWM2B	gpmc_ad9	lcd_data22	mmc1_dat1	mmc2_dat5
14	T11	GPIO0_26	gpmc_ad10	lcd_data21	mmc1_dat2	mmc2_dat6
15	U13	GPIO1_15	gpmc_ad15	lcd_data16	mmc1_dat7	mmc2_dat3
16	V13	GPIO1_14	gpmc_ad14	lcd_data17	mmc1_dat6	mmc2_dat2
17	U12	GPIO0_27	gpmc_ad11	lcd_data20	mmc1_dat3	mmc2_dat7
18	V12	GPIO2_1	gpmc_clk_mux0	lcd_memory_clk	gpmc_wait1	mmc2_clk
19	U10	EHRPWM2A	gpmc_ad8	lcd_data23	mmc1_dat0	mmc2_dat4
20	V9	GPIO1_31	gpmc_csn2	gpmc_be1n	mmc1_cmd	
21	U9	GPIO1_30	gpmc_csn1	gpmc_clk	mmc1_clk	
22	V8	GPIO1_5	gpmc_ad5	mmc1_dat3		
23	U8	GPIO1_4	gpmc_ad4	mmc1_dat4		
24	V7	GPIO1_1	gpmc_ad1	mmc1_dat1		
25	U7	GPIO1_0	gpmc_ad0	mmc1_dat0		
26	V6	GPIO1_29	gpmc_csn0			
27	U5	GPIO2_22	lcd_vsync	gpmc_a8		
28	V5	GPIO2_24	lcd_pclk	gpmc_a10		
29	R5	GPIO2_23	lcd_hsync	gpmc_a9		
30	R6	GPIO2_25	lcd_ac_bias_en	gpmc_a11		
31	V4	UART5_CTSN	lcd_data14	gpmc_a18	eQEP1_index	mcasp0_axr1
32	T5	UART5_RTSN	lcd_data15	gpmc_a19	eQEP1_strobe	mcasp0_ahclkx
33	V3	UART4_RTSN	lcd_data13	gpmc_a17	eQEP1B_in	mcasp0_fsr
34	U4	UART3_RTSN	lcd_data11	gpmc_a15	ehrpwm1B	mcasp0_ahclkr
35	V2	UART4_CTSN	lcd_data12	gpmc_a16	eQEP1A_in	mcasp0_aclkr

PIN	PROC	NAME	MODE0	MODE1	MODE2	MODE3
36	U3	UART3_CTSN	lcd_data10	gpmc_a14	ehrpwm1A	mcasp0_axr0
37	U1	UART5_TXD	lcd_data8	gpmc_a12	ehrpwm1_tripzone_in	mcasp0_aclkx
38	U2	UART5_RXD	lcd_data9	gpmc_a13	ehrpwm0_synco	mcasp0_fsx
39	T3	GPIO2_12	lcd_data6	gpmc_a6		eQEP2_index
40	T4	GPIO2_13	lcd_data7	gpmc_a7		eQEP2_strobe
41	T1	GPIO2_10	lcd_data4	gpmc_a4		eQEP2A_in
42	T2	GPIO2_11	lcd_data5	gpmc_a5		eQEP2B_in
43	R3	GPIO2_8	lcd_data2	gpmc_a2		ehrpwm2_tripzone_in
44	R4	GPIO2_9	lcd_data3	gpmc_a3		ehrpwm0_synco
45	R1	GPIO2_6	lcd_data0	gpmc_a0		ehrpwm2A
46	R2	GPIO2_7	lcd_data1	gpmc_a1		ehrpwm2B

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 10 shows the other **P8** signals for modes 4-7.

Table 10. P8 Mux Options Modes 4-7

PIN	PROC	NAME	MODE4	MODE5	MODE6	MODE7
1		GND				
2		GND				
3	R9	GPIO1_6				gpio1[6]
4	T9	GPIO1_7				gpio1[7]
5	R8	GPIO1_2				gpio1[2]
6	T8	GPIO1_3				gpio1[3]
7	R7	TIMER4				gpio2[2]
8	T7	TIMER7				gpio2[3]
9	T6	TIMER5				gpio2[5]
10	U6	TIMER6				gpio2[4]
11	R12	GPIO1_13	eQEP2B_in			gpio1[13]
12	T12	GPIO1_12	EQEP2A_IN			gpio1[12]
13	T10	EHRPWM2B	ehrpwm2B			gpio0[23]
14	T11	GPIO0_26	ehrpwm2_tripzone_in			gpio0[26]
15	U13	GPIO1_15	eQEP2_strobe			gpio1[15]
16	V13	GPIO1_14	eQEP2_index			gpio1[14]
17	U12	GPIO0_27	ehrpwm0_synco			gpio0[27]
18	V12	GPIO2_1			mcasp0_fsr	gpio2[1]
19	U10	EHRPWM2A	ehrpwm2A			gpio0[22]
20	V9	GPIO1_31				gpio1[31]
21	U9	GPIO1_30				gpio1[30]
22	V8	GPIO1_5				gpio1[5]
23	U8	GPIO1_4				gpio1[4]
24	V7	GPIO1_1				gpio1[1]
25	U7	GPIO1_0				gpio1[0]
26	V6	GPIO1_29				gpio1[29]
27	U5	GPIO2_22				gpio2[22]
28	V5	GPIO2_24				gpio2[24]
29	R5	GPIO2_23				gpio2[23]
30	R6	GPIO2_25				gpio2[25]
31	V4	UART5_CTSN	uart5_rxd		uart5_ctsn	gpio0[10]
32	T5	UART5_RTSN	mcasp0_axr3		uart5_rtsn	gpio0[11]
33	V3	UART4_RTSN	mcasp0_axr3		uart4_rtsn	gpio0[9]
34	U4	UART3_RTSN	mcasp0_axr2		uart3_rtsn	gpio2[17]
35	V2	UART4_CTSN	mcasp0_axr2		uart4_ctsn	gpio0[8]
36	U3	UART3_CTSN			uart3_ctsn	gpio2[16]
37	U1	UART5_TXD	uart5_txd		uart2_ctsn	gpio2[14]

PIN	PROC	NAME	MODE4	MODE5	MODE6	MODE7
38	U2	UART5_RXD	uart5_rxd		uart2_rtsn	gpio2[15]
39	T3	GPIO2_12				gpio2[12]
40	T4	GPIO2_13	pr1_edio_data_out7			gpio2[13]
41	T1	GPIO2_10				gpio2[10]
42	T2	GPIO2_11				gpio2[11]
43	R3	GPIO2_8				gpio2[8]
44	R4	GPIO2_9				gpio2[9]
45	R1	GPIO2_6				gpio2[6]
46	R2	GPIO2_7				gpio2[7]

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.

6.13.3 Expansion Header P9

Table 11 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. Signals highlighted in yellow are changes from the previous revision of the SRM.

Table 11. Expansion Header P9 Pinout

SIGNAL NAME	PIN	CONN		PIN	SIGNAL NAME
	GND	1	2	GND	
	VDD_3V3EXP	3	4	VDD_3V3EXP	
	VDD_5V	5	6	VDD_5V	
	SYS_5V	7	8	SYS_5V	
PWR_BUT*		9	10	A10	SYS_RESETn
UART4_RXD	T17	11	12	U18	GPIO1_28
UART4_TXD	U17	13	14	U14	EHRPWM1A
GPIO1_16	R13	15	16	T14	EHRPWM1B
I2C1_SCL	A16	17	18	B16	I2C1_SDA
I2C2_SCL	D17	19	20	D18	I2C2_SDA
UART2_TXD	B17	21	22	A17	UART2_RXD
GPIO1_17	V14	23	24	D15	UART1_TXD
GPIO3_21	A14	25	26	D16	UART1_RXD
GPIO3_19	C13	27	28	C12	SPI1_CS0
SPI1_D0	B13	29	30	D12	SPI1_D1
SPI1_SCLK	A13	31	32	VDD_ADC(1.8V)	
AIN4	C8	33	34	GNDA_ADC	
AIN6	A5	35	36	A5	AIN5
AIN2	B7	37	38	A7	AIN3
AIN0	B6	39	40	C7	AIN1
CLKOUT2	D14	41	42	C18	GPIO0_7
	GND	43	44	GND	
	GND	45	46	GND	

*PWR_BUT is a 5V level as pulled up internally by the TPS65217B. It is activated by pulling the signal to GND.

6.13.3.1 Connector P9 Signal Pin Mux Options

Table 12 gives the pin mux options for the signals for connector **P9** for modes 0-3.

Table 12. P9 Mux Options Modes 0-3

PIN	PROC	SIGNAL NAME	MODE0	MODE1	MODE2	MODE3
1		GND				
2		GND				
3		DC_3.3V				
4		DC_3.3V				
5		VDD_5V				
6		VDD_5V				
7		SYS_5V				
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
12	U18	GPIO1_28	gpmc_be1n	mii2_col	gpmc_csn6	mmc2_dat3
13	U17	UART4_TXD	gpmc_wpn	mii2_rxerr	gpmc_csn5	rmii2_rxerr
14	U14	EHRPWM1A	gpmc_a2	mii2_txd3	rgmii2_td3	mmc2_dat1
15	R13	GPIO1_16	gpmc_a0	gmii2_txen	rmii2_tctl	mii2_txen
16	T14	EHRPWM1B	gpmc_a3	mii2_txd2	rgmii2_td2	mmc2_dat2
17	A16	I2C1_SCL	spi0_cs0	mmc2_sdwp	I2C1_SCL	ehrpwm0_synci
18	B16	I2C1_SDA	spi0_d1	mmc1_sdwp	I2C1_SDA	ehrpwm0_tripzone
19	D17	I2C2_SCL	uart1_rtsn	timer5	dcan0_rx	I2C2_SCL
20	D18	I2C2_SDA	uart1_ctsn	timer6	dcan0_tx	I2C2_SDA
21	B17	UART2_TXD	spi0_d0	uart2_txd	I2C2_SCL	ehrpwm0B
22	A17	UART2_RXD	spi0_sclk	uart2_rxd	I2C2_SDA	ehrpwm0A
23	V14	GPIO1_17	gpmc_a1	gmii2_rxdv	rgmii2_rxdv	mmc2_dat0
24	D15	UART1_TXD	uart1_txd	mmc2_sdwp	dcan1_rx	I2C1_SCL
25	A14	GPIO3_21	mcasp0_ahclkx	eQEP0_strobe	mcasp0_axr3	mcasp1_axr1
26	D16	UART1_RXD	uart1_rxd	mmc1_sdwp	dcan1_tx	I2C1_SDA
27	C13	GPIO3_19	mcasp0_fsr	eQEP0B_in	mcasp0_axr3	mcasp1_fsx
28	C12	SPI1_CS0	mcasp0_ahclkr	ehrpwm0_synci	mcasp0_axr2	spi1_cs0
29	B13	SPI1_D0	mcasp0_fsx	ehrpwm0B		spi1_d0
30	D12	SPI1_D1	mcasp0_axr0	ehrpwm0_tripzone		spi1_d1
31	A13	SPI1_SCLK	mcasp0_aclkx	ehrpwm0A		spi1_sclk
32		VADC				
33	C8	AIN4				
34		AGND				
35	A8	AIN6				

PIN	PROC	SIGNAL NAME	MODE0	MODE1	MODE2	MODE3
36	B8	AIN5				
37	B7	AIN2				
38	A7	AIN3				
39	B6	AIN0				
40	C7	AIN1				
41	D14	CLKOUT2	xdma_event_intr1		tclkin	clkout2
42	C18	GPIO0_7	eCAP0_in_PWM0_out	uart3_txd	spi1_cs1	pr1_ecap0_ecap_capin_apwm_o
43		GND				
44		GND				
45		GND				
46		GND				

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 13 gives the pin mux options for the signals for connector **P9** for modes 4-7.

Table 13. P9 Mux Options Modes 4-7

PIN	PROC	SIGNAL NAME	MODE4	MODE5	MODE6	MODE7
1		GND				
2		GND				
3		DC_3.3V				
4		DC_3.3V				
5		VDD_5V				
6		VDD_5V				
7		SYS_5V				
8		SYS_5V				
9		PWR_BUT				
10	A10	SYS_RESETn				
11	T17	UART4_RXD	mmc1_sdcd		uart4_rxd_mux2	gpio0[30]
12	U18	GPIO1_28	gpmc_dir		mcasp0_aclkr_mux3	gpio1[28]
13	U17	UART4_TXD	mmc2_sdcd		uart4_txd_mux2	gpio0[31]
14	U14	EHRPWM1A	gpmc_a18		ehrpwm1A_mux1	gpio1[18]
15	R13	GPIO1_16	gpmc_a16		ehrpwm1_tripzone_input	gpio1[16]
16	T14	EHRPWM1B	gpmc_a19		ehrpwm1B_mux1	gpio1[19]
17	A16	I2C1_SCL				gpio0[5]
18	B16	I2C1_SDA				gpio0[4]
19	D17	I2C2_SCL	spi1_cs1			gpio0[13]
20	D18	I2C2_SDA	spi1_cs0			gpio0[12]
21	B17	UART2_TXD			EMU3_mux1	gpio0[3]
22	A17	UART2_RXD			EMU2_mux1	gpio0[2]
23	V14	GPIO1_17	gpmc_a17		ehrpwm0_synco	gpio1[17]
24	D15	UART1_TXD				gpio0[15]
25	A14	GPIO3_21	EMU4_mux2			gpio3[21]
26	D16	UART1_RXD				gpio0[14]
27	C13	GPIO3_19	EMU2_mux2			gpio3[19]
28	C12	SPI1_CS0	eCAP2_in_PWM2_out			gpio3[17]
29	B13	SPI1_D0	mmc1_sdcd_mux1			gpio3[15]
30	D12	SPI1_D1	mmc2_sdcd_mux1			gpio3[16]
31	A13	SPI1_SCLK	mmc0_sdcd_mux1			gpio3[14]
32		VDD_ADC (1.8V_				
33	C8	AIN4				
34		GNDA_ADC				
35	A5	AIN6				

PIN	PROC	SIGNAL NAME	MODE4	MODE5	MODE6	MODE7
36	A5	AIN5				
37	B7	AIN2				
38	A7	AIN3				
39	B6	AIN0				
40	C7	AIN1				
41	D14	CLKOUT2	timer7_mux1		EMU3_mux0	gpio0[20]
42	C18	GPIO0_7	spi1_sclk	mmc0_sdwp	xdma_event_intr2	gpio0_7
43		GND				
44		GND				
45		GND				
46		GND				

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.

6.13.4 PMIC Expansion Header

There is an additional connector that brings out some additional signals from the **TPS65217B** power management chip. **Figure 25** shows the PMIC expansion connector.

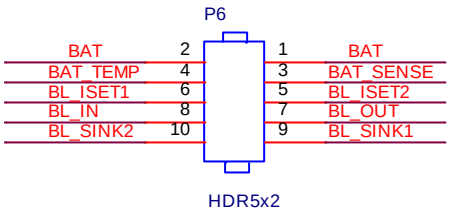


Figure 26. PMIC Expansion Header

6.13.5 Backlight Interface

The most useful interface provided is the backlight interface which is very useful for powering the backlight of LCD panels. The Backlight circuit is a boost converter and two current sinks capable of driving up to 2x10 LEDs at 25mA or a single string at 50mA of current. Two current levels can be programmed using two external resistors and brightness dimming is supported by an internal PWM signal under I2C control. Both current sources are controlled together and cannot operate independently. The boost output voltage is internally limited to 39V. LED current is selected through the ISEL bit of the same register as is the PWM frequency. By default, the PWM frequency is set to 200Hz but can be changed to 100Hz, 500Hz, and 1000Hz. The PWM duty cycle can be adjusted from 1% to 100% in 1% steps through the WLEDCTRL2 register. If only a single WLED string is required, short both ISINK pins together and connect them to the Cathode of the diode string. Note that the LED current in this case is doubled and to compensate, the RSET resistors must be doubled as well. **Figure 26** below shows the two different circuits.

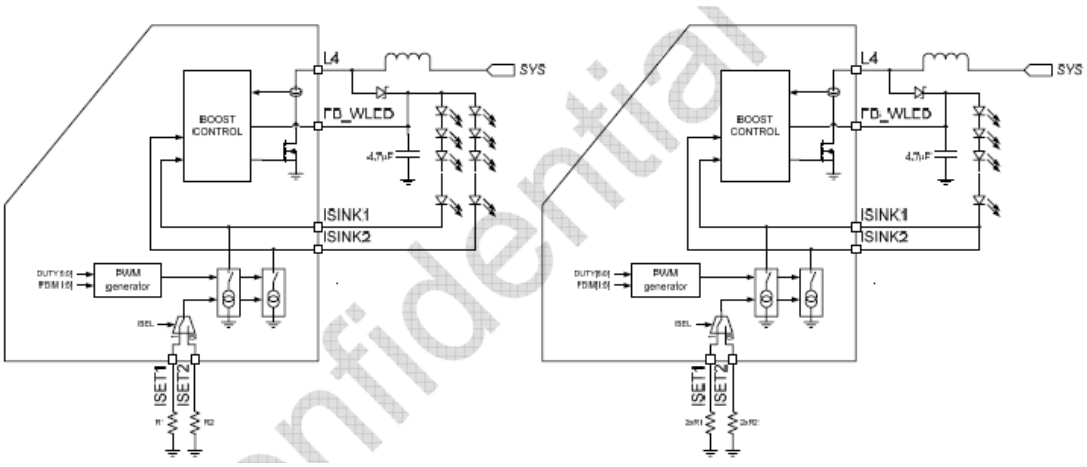


Figure 27. Backlight Circuitry

For more information on working with this interface, refer to the **TPS65217B** datasheet.

6.13.6 Battery Interface

There is also a battery charger interface. This interface can be used by anyone wanting to experiment with batteries and battery charging. However, as a source for powering the BeagleBone, this interface is not practical as there is no way to provide the 5V on the board required for the USB host port. The reason for this is that the maximum battery voltage is 3.7V, well short of 5V. The LDOs on the **TPS65217B** are 200mv, meaning that the 3.7V battery LDOS can supply the needed 3.3V after the battery starts discharging as long as it does not go below 3.5V, including any voltage drop for the connections that may occur. If you are OK with not having a USB host function, then it is possible to use the battery charger for the purpose of a battery powered system. If you have an LCD, then most of the LCD Capes do require 5V as well to operate. This limits the application for battery power as the BeagleBone is currently designed. There are no plans to add an extra switcher on the BeagleBone to boost the 3.7V to 5V for this issue.

Figure 27 shows the battery circuitry inside the **TPS65217B**.

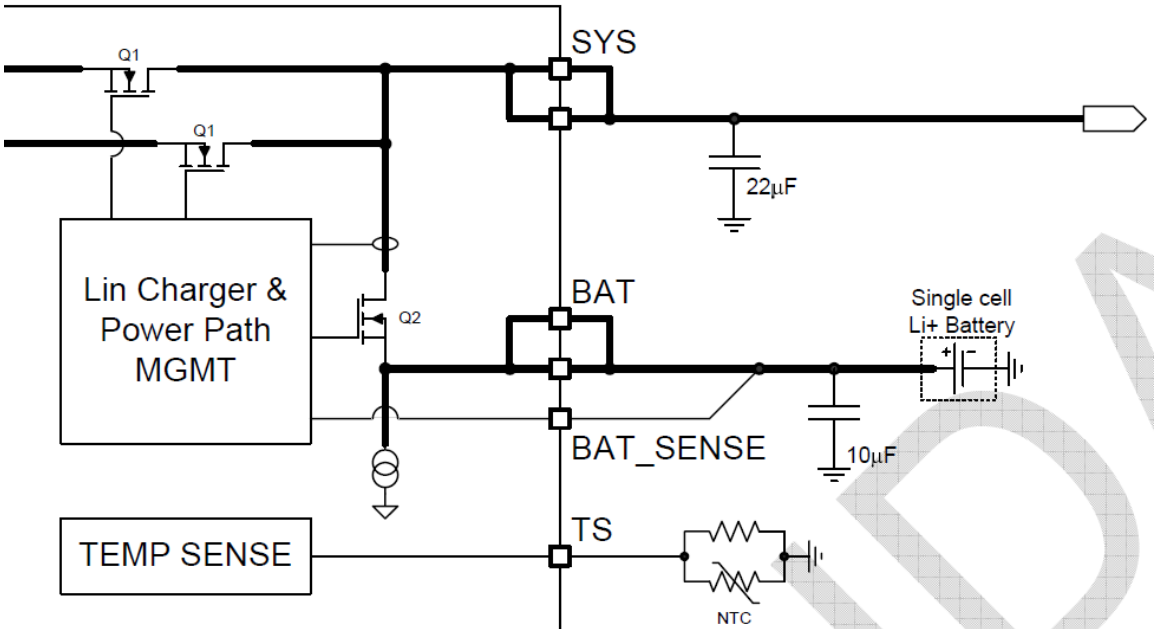


Figure 28. Battery Circuitry

7.0 Cape Board Support

The BeagleBone 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 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. 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.

7.1 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.

The address of the EEPROM will be set via either jumpers or a dipswitch on each expansion board. **Figure 28** below is the design of the EEPROM circuit.

The EEPROM used is the same one as is used on the BeagleBone, 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.

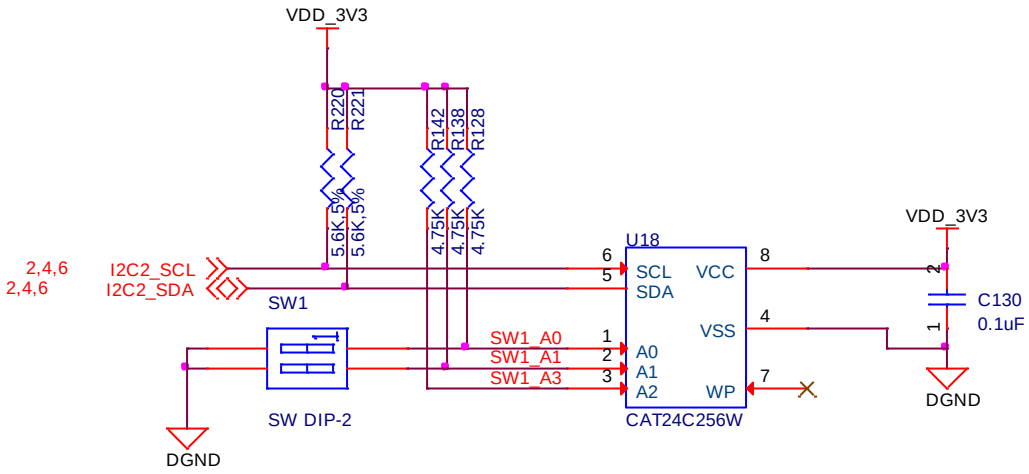


Figure 29. Expansion Board EEPROM No Write Protect

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.

7.1.1 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.

7.1.2 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.

7.1.3 EEPROM Write Protect

The design in **Figure 28** 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 29** 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.

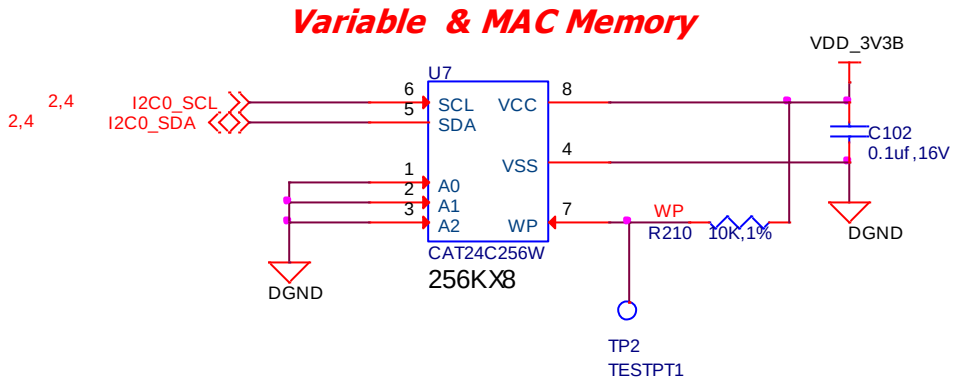


Figure 30. Expansion Board EEPROM Write Protect

7.1.4 EEPROM Data Format

Table 14 below 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 14. 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	60	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: WWYY&&&&nnnn where: WW = 2 digit week of the year of production YY = 2 digit year of production &&&&=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. 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 MSB LSB Bit order: 15 14 ……1..0 Bit 15……..Pin is used or not……...0=Unused by Cape 1=Used by Cape Bit 14-13……Pin Direction……..….1 0=Output 01=Input 11=BDIR Bits 12-7……Reserved Bit 6……..Slew Rate ……..0=Fast 1=Slow Bit 5…….……Rx Enable……..0=Disabled 1=Enabled Bit 4……..…Pull Up/Dn Select……..0=Pulldown 1=PullUp Bit 3……..Pull Up/DN enabled……..0=Enabled 1=Disabled 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.

7.1.5 Pin Usage

Table 15 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 15. 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																

7.2 Pin Usage Consideration

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

7.2.1 Boot Pins

There are 16 pins that control the boot mode of the processor that are exposed on the expansion headers. **Figure 31** below shows those signals:

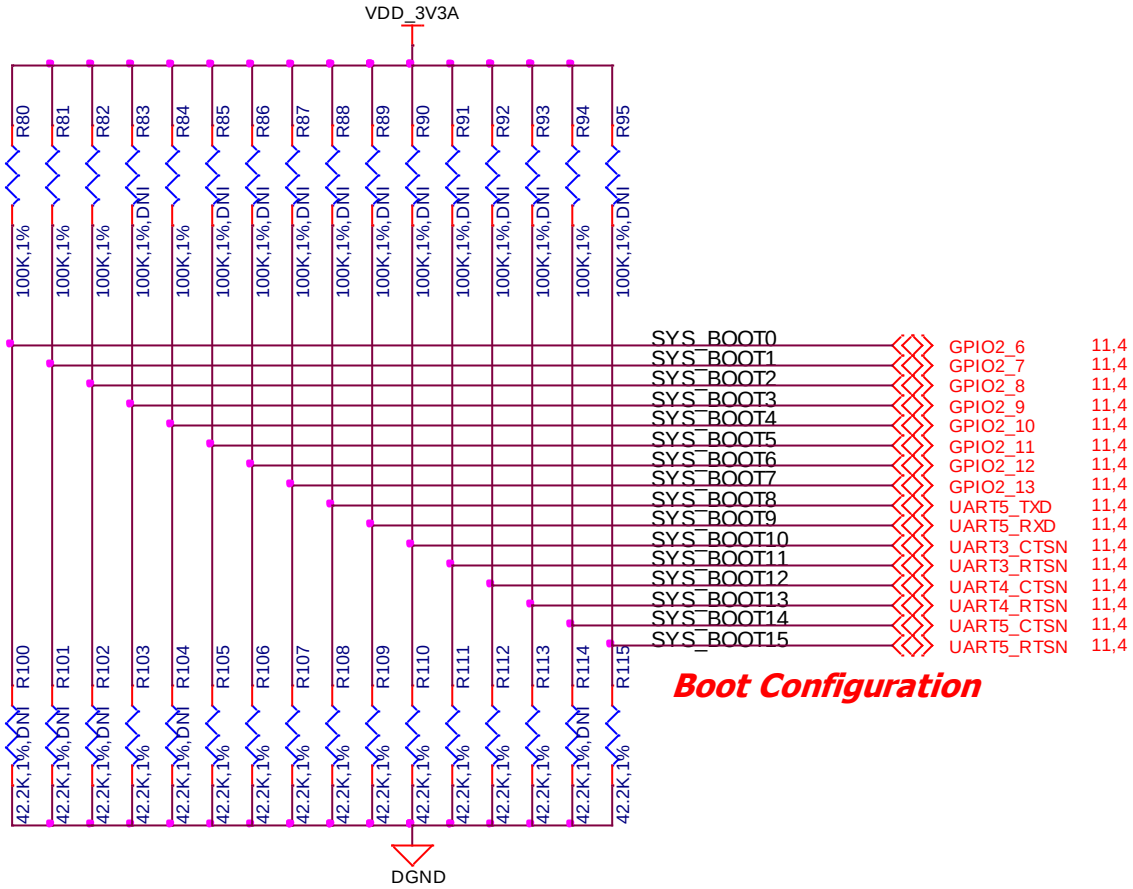


Figure 31. Expansion Boot Pins

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.

7.3 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.

NOTE: ***Be careful if you are considering using standoffs on the BeagleBone Rev A3 A4 or A5. The mounting hole next to the DC power jack has resistors that are a little too close to the hole and if you are not careful, you can damage those resistors when attaching the standoff. Use as small a diameter standoff as possible This issue has been resolved on the Rev A6 version Typically the retention force of the expansion headers is enough to secure the boards and standoffs are not needed..***

7.3.1 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 29** is a picture of the connector. These are dual row 23 position 2.54mm x 2.54mm connectors.

Figure 32. Single Expansion Connector

The connector is typically mounted on the bottom side of the board as shown in **Figure 30**. 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.

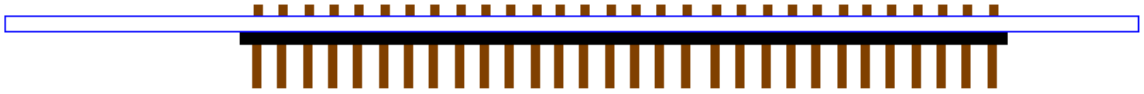


Figure 33. Single Cape Expansion Connector

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 15** 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 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 15** 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.

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

Table 16. 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

7.3.2 Battery Connector- Single

For non-stacking or single configuration this connector is a single 10 pin expansion header. **Figure 31** 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 34. Battery/Backlight Expansion Connector

Table 18 below is the possible part numbers for this connector. The first item in **Table 17** 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.

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

Table 17. 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

7.3.3 Main Expansion Headers-Stacking

For stacking configuration, the two 46 pin expansion headers use the same connectors. **Figure 32** is a picture of the connector. These are dual row 23 position 2.54mm x 2.54mm connectors.

Figure 35. Expansion Connector

The connector is mounted on the top side of the board with longer tails to allow insertion into the BeagleBone. **Figure 33** is the connector configuration for the connector.

Figure 36. Stacked Cape Expansion Connector

For convenience listed in **Table 17** 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 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 18** 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.

Please refer to **Section 8.3** for more information on the connectors and the insertion force issue. The third part listed in **Table 18** will have insertion force issues.

Table 18. 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.

7.3.4 Battery Connector Stacking

This connector is a single two 10 pin expansion header. **Figure 34** is a picture of the connector. This is a dual row 10 position 2.54mm x 2.54mm connector and is the same as the main connector except with less positions.

Figure 37. Stacked Battery 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. . The first item in **Table 19** 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.

Please refer to **Section 8.3** for more information on the connectors and the insertion force issue. The third part listed in **Table 19** will have insertion force issues.

Table 19. Stacked Cape Connectors

SUPPLIER	PARTNUMBER	TAIL LENGTH	OVER HANG
Major League	SSHQ-105-D-06-GT-LF	.190	0.049
Major League	SSHQ-105-D-08-GT-LF	.390	0.249
Major League	SSHQ-105-D-10-GT-LF	.560	0.419

Tail length does not include the thickness of the Cape PCB.

7.3.5 Stacked Capes w/Signal Stealing

Figure 35 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.

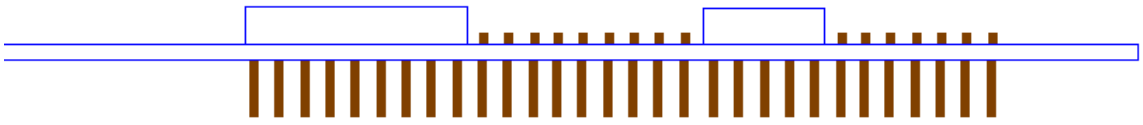


Figure 38. Stacked w/Signal Stealing Expansion Connector

7.3.6 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. 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, 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.

7.3.7 BeagleBone Female Connectors

Figure 36 below shows the key measurements used in calculating how much the pin extends past the contact point on the connector, what we call overhang.

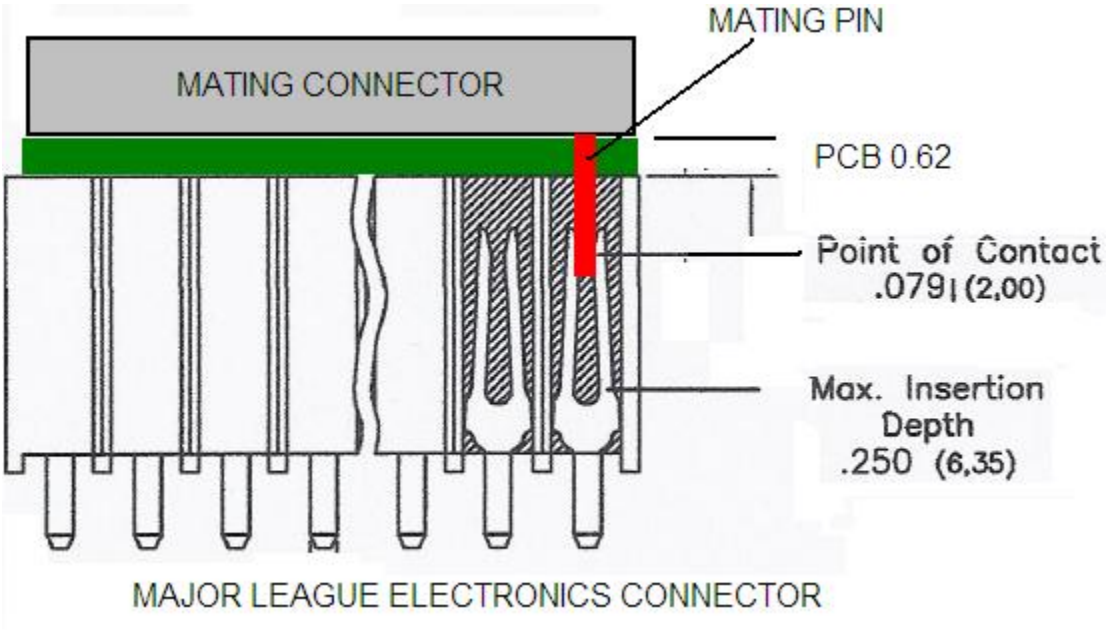


Figure 39. Connector Pin Insertion Depth

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.

7.4 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.

7.5 Cape Power

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

7.5.1 Main Board Power

The **Table 19** 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 20. 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 and is the primary power rail for expansion boards.

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.

7.5.2 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.

7.6 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. 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 board.

7.6.1 Standard Cape Size

Figure 40 is the outline of the standard Cape. The dimensions are in inches.

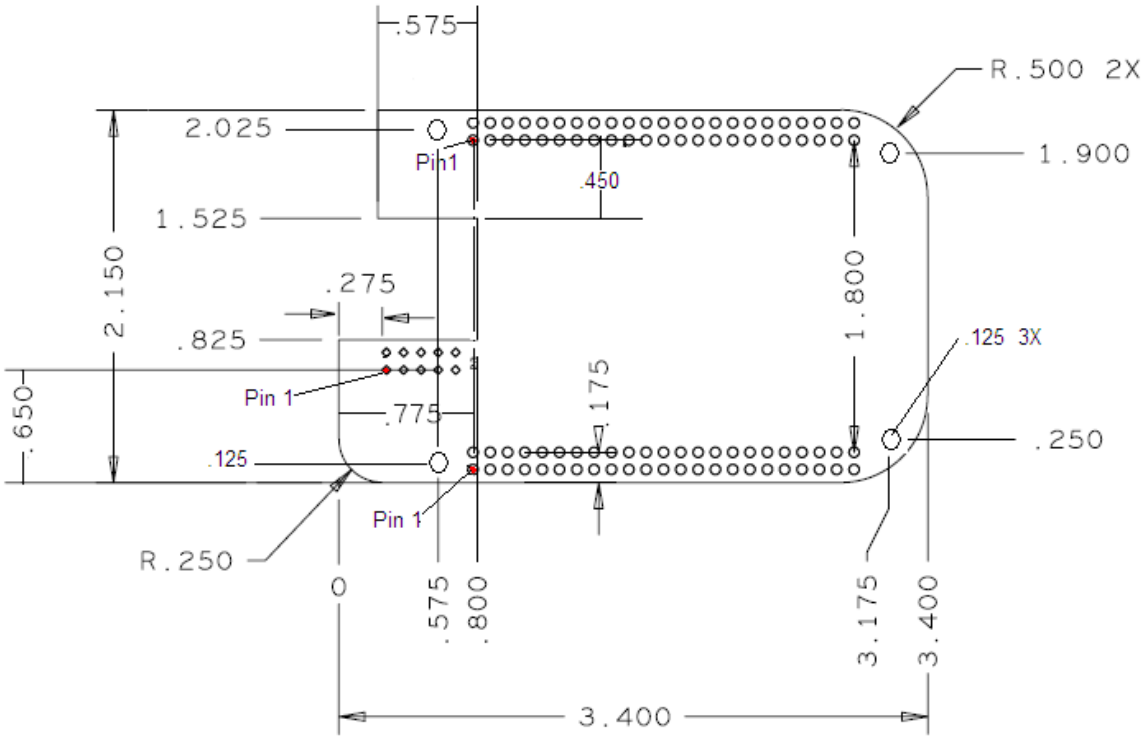


Figure 40. Cape Board Dimensions

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 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 board in this respect.

7.6.2 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 is not plugged in incorrectly in such a manner that damage would be cause to the BeagleBone 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.

7.6.3 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.

8.0 Board Setup

This section describes how to setup the board and to make sure that it is operating. It also provides an advanced section that allows you to run a self diagnostic test that does require additional equipment to be purchased.

8.1 Creating A SD Card

If you need to create an SD card for the board that is the same as what ships with the BeagleBone, you can follow the instructions found at the following location:

http://circuitco.com/support/index.php?title=BeagleBone

Other methods are also possible if you are familiar with Linux. Instructions are found at the following link which also will have the latest image.

http://www.angstrom-distribution.org/demo/beaglebone/

You will need a 4GB microSD card.

8.2 USB Powered Setup

The board ships with everything you need for this configuration.

- BeagleBone
- microSD card with bootable SW
- USB Type A to 5 pin connector

To setup the board:

1) Insert the SD card into the SD card connector
2) Plug the USB cable into the BeagleBone
3) Plug the other end of the USB cable into the PC USB port.
4) The power LED D1 should be on
5) After a few seconds, USER0 and USER1 LED should start flashing
6) After 10 seconds or so, the board should show up as a mass storage device on your PC
7) Open the new drive and click on the Readme.html file.
8) The file should open in your browser.
9) Follow the instructions on the HTML page.

8.3 DC Powered Setup

The board ships with everything you need for this configuration except for a power supply. The first three items below are provided and the power supply will need to be provided by you.

- BeagleBone
- microSD card with bootable SW
- USB Type A to 5 pin connector
- 5VDC 1A power supply w/2.1mm x 5.5mm connector, center positive.

To setup the board:

1) Insert the SD card into the SD card connector
2) Plug the DC cable into the board.
3) The power LED D1 should be on
4) Plug the USB cable into the BeagleBone
5) Plug the other end of the USB cable into the PC USB port.
6) After a few seconds, USER0 and USER1 LED should start flashing
7) After 10 seconds or so, the board should show up as a mass storage device on your PC
8) Open the new drive and click on the Readme.html file.
9) The file should open in your browser.
10) Follow the instructions on the HTML page.

8.4 Advanced Test

This test involves the purchase of a USB hub that is equipped with an Ethernet port or the use of a USB Hub with a USB to Ethernet Dongle plugged in. The SW that ships with the board is capable of running this test. You may need to load drivers for your particular Hub or Ethernet dongle.

The following procedure will setup and test the board. The following items are tested on the board:

- USB Client Port
- USB Host Port
- Ethernet Port
- DDR
- PMIC
- EEPROM
- Processor
- SD Slot
- DC Power
- USB HUB

- USB to Serial
- LEDs

8.4.1 Equipment Needed

The following items are needed to perform this test:

1) USB Hub with Ethernet port
2) Ethernet Cable
3) USB A Male to 5pin male
4) BeagleBone
5) 26 AWG jumper wire, stripped
6) 5VDC 1A power supply, 2.1mm Center positive

8.4.2 Procedure

1) Connect the USB HUB to the USB Host port of the BeagleBone.
2) Connect the HUB Ethernet port to the BeagleBone Ethernet port.
3) Connect one of the USB ports to USB connector on the BeagleBone.
4) Insert the SD card that came with the board into the SD connector.
5) Add a jumper wire between pin 2 and 3 of the P8. This tells the SW to run the test.
6) Insert the DC power supply
7) The PWR LED should turn on.
8) Then D2 and D3 should start flashing indicating the boot process has begun.
9) After about a minute, D2 and D3 should turn off and D5 should start flashing.
10) D5 will continue to flash during the test process which should take about 2-3 minutes.
11) At the end of the test one of two things will happen:
    a. If all the LEDS are on solid, then the board has passed the test.
    b. If all LEDS are flashing, then the board has failed the test.

8.4.3 Debugging

It is possible to add a USB to serial cable to the external HUB for messages as the tests run. This will tell you where the test fails. It will require a USB to serial adapter to also be plugged into your PC and a Null modem female to female adapter be placed between the two cables.

In order for this to work, the Linux driver needs to be installed on the BeagleBone for the USB to serial adapter. For now, only one USB to serial adapter is supported. Others will be added over time.

Once you have the correct cable configuration, you can open up a terminal program set to the serial port and set for 115KBaud, 8,n,1 and no handshaking. The results of the test as run will be printed to the terminal.

9.0 Software Support

This section provides assistance in working with the Software that comes with the Beaglebone. The primary support mailing list is duscuss@beagleboard.org

9.1 Tutorials

Have a look at these websites to get an idea of what people have been working on. It should prove helpful to you.

The Ångström website has links to various tutorials and projects, you can find it at http://www.angstrom-distribution.org/

Limor Fried of adafruit.com fame has started a collection of Beaglebone related tutorials of one which deals with wifi:
http://ladyada.net/products/beaglebone/index.html

Dan Watts has a number of tutorial on how to use the GPIOs and PWM pins:
http://www.gigamegablog.com/tag/beaglebone/

Graeme Gregory has published an example kernel development workflow:
http://www.slimlogic.co.uk/2011/05/openembeddedangstrom-kernel-workflow/

9.2 Reinstalling The Angstrom Image

To reinstall the SD card image you can completely reimage the SD card using Linux:

Visit http://downloads.angstrom-distribution.org/demo/beaglebone/ and get the *img.xz* file you want. Then find out which drive corresponds with your SD card reader:

```
$ dmesg | grep sd
```

The output should look something like this:

```
sd 8:0:0:0: Attached scsi generic sg4 type 0
sd 8:0:0:0: [sde] 7626752 512-byte logical blocks: (3.90 GB/3.63 GiB)
sd 8:0:0:0: [sde] Write Protect is off
```

```
sd 8:0:0:0: [sde] Mode Sense: 03 00 00 00
sd 8:0:0:0: [sde] No Caching mode page present
sd 8:0:0:0: [sde] Assuming drive cache: write through
sd 8:0:0:0: [sde] No Caching mode page present
sd 8:0:0:0: [sde] Assuming drive cache: write through
sde: sde1 sde2
```

This shows that it detected a 4GB card and assigned it to **/dev/sde**. In the next steps replace **/dev/sdX** with the name from the previous step. Be **very** careful with this. Using the wrong name can result in an erased hard drive.

To reimage the SD card do the following:

```
$ sudo -s
  (type in your password)
 # xz -dkc imagename.img.xz > /dev/sdX
 # exit
```

This will take more than 20 minutes, usually 45-60 minutes, depending on the speed of your card reader and SD card.

9.3 Rebuilding The Angstrom Image

The SD card image in the box is based on the Ångström distribution. All Ångström binaries are built using OpenEmbedded. This section describes the steps necessary to setup an environment where you can rebuild the images and packages yourself.

The build is managed by scripts to make things easier, so get the setup scripts:

```
$ git clone git://github.com/Angstrom-distribution/setup-scripts.git
```

If you are behind a firewalling proxy, have a look at the oebb.sh file, it has built-in proxy handling.

Configure the setup scripts for the beaglebone:

```
$ MACHINE=beaglebone ./oebb.sh config beaglebone
```

Start with a kernel build:

```
$ MACHINE=beaglebone ./oebb.sh bitbake virtual/kernel
```

Or a small command line image:

```
$ MACHINE=beaglebone./oebb.sh bitbake systemd-image
```

Or rebuild the SD card image:

```
$ MACHINE=beaglebone ./oebb.sh bitbake cloud9-gnome-image
```

10.0 BeagleBone Mechanical Specification

Size:	3.5” x 2.1” (86.36mm x 53.34mm)
Max height:	.187” (4.76mm)
PCB Layers:	6
PCB thickness:	.062”
RoHS Compliant:	Yes
Weight:	1.4 oz

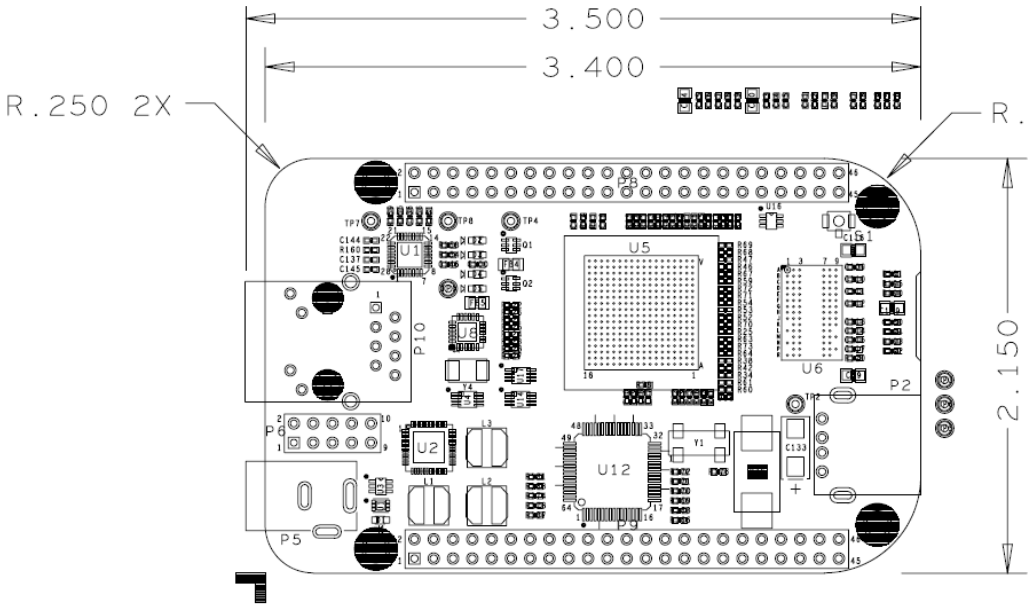


Figure 41. Board Top Profile

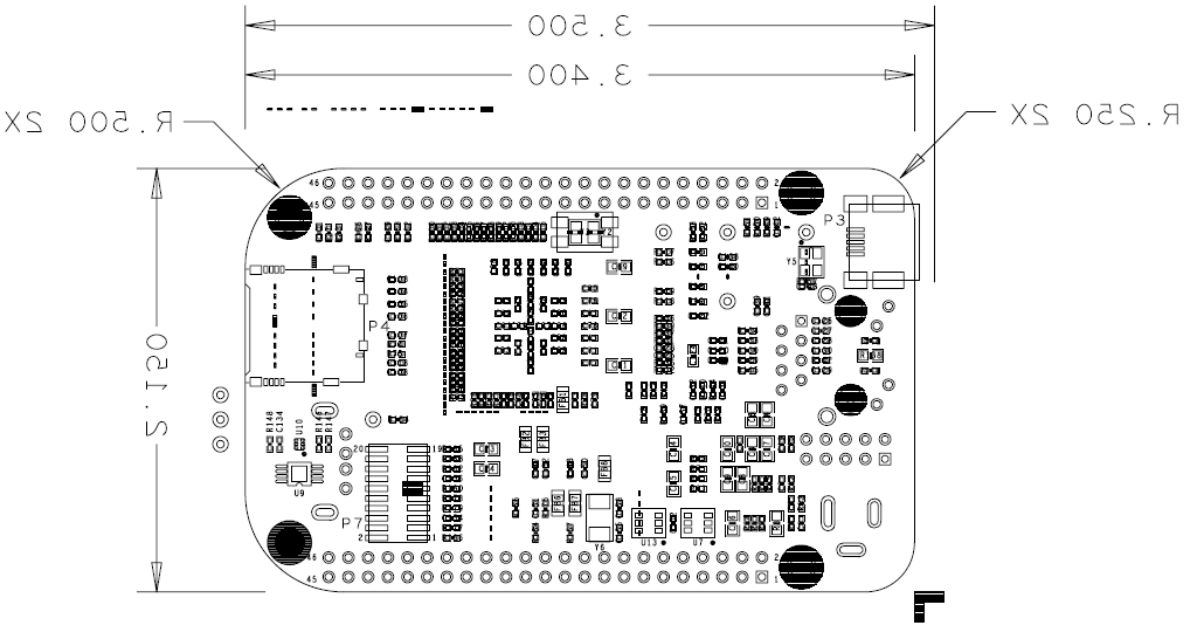


Figure 42. Board Bottom Profile

11.0 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=BeagleBone

ALL support for this design is through the BeagleBoard.org community at beagleboard@googlegroups.com .

There are also some community members working to convert the schematics and PCB files into other formats. Look for those to available in the future.