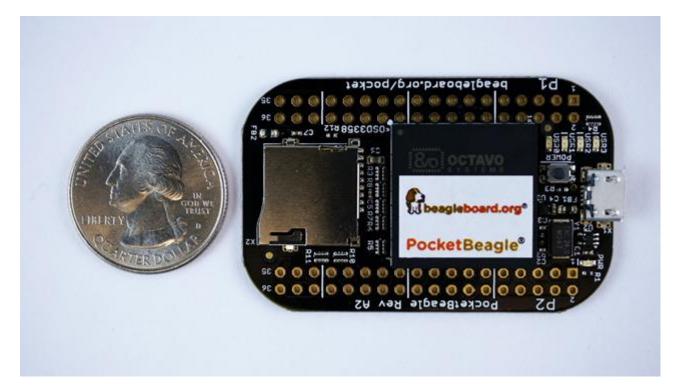
## System Reference Manual

Jason Kridner edited this page on Jan 15  $\cdot$  <u>118 revisions</u>



#### PocketBeagle

System Reference Manual (SRM)

**Revision A.x (on-line wiki edition)** 

December 6, 2017

Maintaining author: Jason Kridner jkridner@beagleboard.org

#### Contributing Editor: Cathy Wicks cathy@beagleboard.org

#### THIS DOCUMENT

**Terms** These design materials are \*NOT SUPPORTED\* and \*DO NOT\* constitute a reference design. Only "community" support is allowed via resources at <u>BeagleBoard.org/discuss</u>.

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.

In other words, you may use the design materials as you choose and there is no license with regards to usage in the manufacturing process. We mean it, these design materials may be totally unsuitable for any purposes. Don't blame us!

As a general rule, we don't encourage use of this or other off-the-shelf single board computers in commercial products without engaging with a manufacturer to create a supplier agreement and make sure that you can get material as your business demands. Further, we do update the design on occasions where we find it necessary and won't guarantee a supply of older revisions, though we do seek periodic manufacturing of all of our boards for a period of roughly 10 years and will make design changes to replace obsolete parts and that may impact your usage. If you do opt to use it in a product, you take full responsibility for that product.

Do not use the BeagleBoard.org logo or trademarks (such as BeagleBoard, BeagleBone and PocketBeagle) on your products without a *logo license* from the BeagleBoard.org Foundation, but feel free to reference BeagleBoard.org.

See the *LICENSE* file regarding the copyright of these materials.

ab	le of Contents
	1.0 Introduction
•	<ul><li>Introduction</li><li>Figure 1. PocketBeagle Home Page</li></ul>
	2.0 Change History
-	2.1 Document Change History
	<u>Table 1. Change History</u>
	<u>2.2 Board Changes</u>
	<u>Table 2. Board History</u>
	<ul> <li><u>2.2.1 PocketBone</u></li> </ul>
	• <u>2.2.2 Rev A1</u>
	• <u>2.2.3 Rev A2</u>
	3.0 Connecting Up PocketBeagle
	<ul> <li><u>3.1 What's In the Package</u></li> <li>Figure 2. PocketBeagle Package</li> </ul>
	<ul> <li>Figure 3. PocketBeagle Package Insert</li> </ul>
	<ul> <li><u>3.2 Connecting the board</u></li> <li><u>2.3 Tethered to a PC using Debian Images</u></li> </ul>
	<ul> <li><u>3.3 Tethered to a PC using Debian Images</u></li> <li>Figure 4. Tethered Configuration</li> </ul>
	<ul> <li>3.3.1 Getting Started</li> </ul>
	<u>Figure 5. Getting Started Page</u>
	<ul> <li>Figure 6. Download Latest Software Image</li> </ul>
	<ul> <li>Figure 7. Download Etcher SD Card Utility</li> </ul>
	<ul> <li>Figure 8. Select the PocketBeagle Image</li> </ul>
	• Figure 9. Burn the Image to the SD Card
	• Figure 10. Insert the microSD Card into PocketBeagle
	• Figure 11. Insert the micro USB Connector into
	PocketBeagle
	Figure 12. Insert the USB connector into PC
	• Figure 13. Board Power LED
	• Figure 14. User LEDs
	<ul> <li>3.3.2 Accessing the Board and Getting Started with Coding</li> </ul>
	<ul> <li>Figure 15. Interactive Quick Start Guide Launch</li> </ul>

- Figure 16. Enable a Network Connection
- Figure 17. Launch Cloud9 IDE
- 3.3.3 Powering Down
  - Figure 20. Power Button
- <u>3.4 Other ways to Connect up to your PocketBeagle</u>

#### 4.0 PocketBeagle Overview

- 4.1 PocketBeagle Features and Specification
  - Table 3. PocketBeagle Features
  - 4.1.1 OSD3358-512M-BSM System in Package
- <u>4.2 Board Component Locations</u>
  - Figure 21. Key Board Component Locations

### 5.0 PocketBeagle High Level Specification

- <u>5.1 Block Diagram</u>
  - Figure 22. PocketBeagle Key Components
- <u>5.2 System in Package (SiP)</u>
  - Figure 23. OSD335x SIP Block Diagram
- <u>5.3 Connectivity</u>
  - <u>5.3.1 Expansion Headers</u>
    - Figure 24. PocketBeagle Expansion Headers
  - <u>5.3.2 microSD Connector</u>
    - Figure 25. microSD Connector
  - <u>5.3.3 USB 2.0 Connector</u>
    - <u>Figure 26. USB 2.0 Connector</u>
  - <u>5.3.4 Boot Modes</u>
    - Table 4. UART Pins on Expansion Headers for Serial Boot
- <u>5.4 Power</u>
- <u>Table 5. Power Inputs Available on Expansion Headers</u>
- <u>Table 6. Power Outputs Available on Expansion Headers</u>
- <u>Table 5. Ground Pins Available on Expansion Headers</u>
- <u>5.5 JTAG Pads</u>
- Figure 27. JTAG Pad Connections
- <u>5.6 Serial Debug Port</u>
  - Figure 28. Serial Debug Connections

### 6.0 Detailed Hardware Design

- 6.1 OSD3358-SM SiP Design
  - <u>6.1.1 SiP A OSD3358 SiP System and Power Signals</u>
    - Figure 29. SiP A OSD3358 SiP System and Power Signals
  - <u>6.1.2 SiP B OSD3358 SiP JTAG, USB & Analog Signals</u>
    - Figure 30. SiP B OSD3358 SiP JTAG, USB & Analog Signals
  - <u>6.1.3 SiP C OSD3358 SiP Peripheral Signals</u>
    - Figure 31. SiP C OSD3358 SiP Peripheral Signals
  - <u>6.1.4 SiP D OSD3358 SiP System Boot Configuration</u>
    - Figure 32. SiP D OSD3358 SiP System Boot Configuration
  - <u>6.1.5 SiP E OSD3358 SiP Power Signals</u>
    - Figure 33. SiP E OSD3358 SiP Power Signals
  - <u>6.1.6 SiP F OSD3358 SiP Power Signals</u>
    - Figure 34. SiP F OSD3358 SiP Power Signals
- <u>6.2 MicroSD Connection</u>

- Figure 35. microSD Connections
- <u>6.3 USB Connector</u>
  - Figure 36. USB Connection
- <u>6.4 Power Button Design</u>
  - Figure 37. Power Button
- <u>6.5 User LEDs</u>
- Figure 38. User LEDs
- Table 6. User LED Control Signals/Pins
- <u>6.6 JTAG Pads</u>
- Figure 39. JTAG Pads Design
- <u>6.7 PRU-ICSS</u>
  - <u>6.7.1 PRU-ICSS Features</u>

.

- 6.7.2 PRU-ICSS Block Diagram
  - Figure 40. PRU-ICSS Block Diagram
- 6.7.3 PRU-ICSS Pin Access
  - Table 6. PRU0 and PRU1 Access

#### 7.0 Connectors

- 7.1 Expansion Header Connectors
  - Figure 41. Expansion Headers for PocketBeagle
  - <u>Figure 42. Expansion Header Popular Functions Color</u> <u>Coded</u>
- <u>7.1.1 P1 Header</u>
- Figure 43 P1 Header
- Table 7. P1 Header Pinout
- <u>7.1.2 P2 Header</u>
- Figure 44. P2 Header
- Table 8. P2 Header Pinout
- <u>7.2 mikroBUS socket connections</u>
  - Figure 45. mikroBUS
  - Figure 46. PocketBeagle Both Headers
- <u>7.3 Setting up an additional USB Connection</u>
- <u>8.0 PocketCape Support</u>
- <u>9.0 PocketBeagle Mechanical</u>
  - <u>9.1 Dimensions and Weight</u>
- 10.0 Additional Pictures
- Figure 47. PocketBeagle Front BW
- <u>Figure 48. PocketBeagle Back BW</u>
- **<u>11.0 Support Information</u>** 
  - <u>11.1 Hardware Design</u>
  - <u>11.2 Software Updates</u>
  - <u>11.5 Export Information</u>
  - <u>11.4 RMA Support</u>
  - <u>11.5 Getting Help</u>

## **1.0 Introduction**

This document is the **System Reference Manual** for PocketBeagle and covers its use and design. PocketBeagle is an ultra-tiny-yet-complete Linux-enabled, community-supported, open-source USB-key-fobcomputer. PocketBeagle features an incredible low cost, slick design and simple usage, making it the ideal development board for beginners and professionals alike. Simply develop directly in a web browser providing you with a playground for programming and electronics. Exploring is made easy with several available libraries and tutorials with many more coming.

PocketBeagle will boot directly from a microSD card. Load a Linux distribution onto your card, plug your board into your computer and get started. PocketBeagle runs GNU.Linux, so you can leverage many different high-level programming languages and a large body of drivers that prevent you from needing to write a lot of your own software.

This design will keep improving as the product matures based on feedback and experience. Software updates will be frequent and will be independent of the hardware revisions and as such not result in a change in the revision number of the board. A great place to find out the latest news and projects for PocketBeagle is on the home page *beagleboard.org/pocket* 

#### https://beagleboard.org/pocket f 💟 in 🛗 8\* eboard.ora Start U Discover Boards # Learn U Explore # Collaborate # BeagleBoard.org > pocket **PocketBeagle**® What is PocketBeagle? PocketBeagle is an ultra-tiny-yet-complete open-source USB-key-fob computer PocketBeagle features an incredible low cost, slick design and simple usage, making PocketBeagle the ideal development board for beginners and professionals alike. Processor: Octavo Systems OSD3358 Summary of Technical Specifications 1GHz ARM® Cortex-A8 Based on new Octavo Systems OSD3358-SM 21mm x 21mm system-in-package that 512MB DDR3 RAM integrated Includes 512MB DDR3 RAM, 1-GHz ARM Integrated power management Cortex-A8 CPU, 2x 200-MHz PRUs, ARM Documentation @» 2×32-bit 200-MHz programmable real-time Cortex-M3, 3D accelerator, power/battery units (PRUs) management and EEPROM ARM Cortex-M3 · 72 expansion pin headers with power and Summary of Key Features Quick Start » battery I/Os, high-speed USB, 8 analog inputs, 44 digital I/Os and numerous digital interface · Low cost Linux computer with tremendous peripherals expansibility Latest Software Images » microUSB host/client and microSD connectors · Opportunity to learn many programming aspects from educators on-line · Openness and flexibility tear-down limits on Subscribe to Newsletter » your imagination Purchase 📜 Select a distributor to buy \*

Figure 1. PocketBeagle Home Page

Make sure you check the support Wiki frequently for the most up to date information.*github.com/beagleboard/pocketbeagle/wiki* 

# 2.0 Change History

This section describes the change history of this document and board. Document changes are not always a result of a board change. A board change will always result in a document change.

## 2.1 Document Change History

Table 1. Change History

Rev	Changes	Date	By
A.x	Production Document	December 7, 2017	JK

## 2.2 Board Changes

#### Table 2. Board History

Rev	Changes	Date	By
A1	Preliminary	February 14, 2017	JK
A2	Production. Fixed mikroBUS Click reset pins (made GPIO).	September 22, 2017	JK

#### 2.2.1 PocketBone

Upon the creation of the first, 27mm-by-27mm, Octavo Systems OSD3358 SIP, Jason did a hack two-layer board in EAGLE called "PocketBone" to drop the Beagle name as this was a totally unofficial effort not geared at being a BeagleBoard.org Foundation project. The board never worked because the 32kHz and 24MHz crystals were backwards and Michael Welling decided to pick it up and redo the design in KiCad as a four-layer board. Jason paid for some prototypes and this resulted in the first successful "PocketBone", a fully-open-source 1-GHz Linux computer in a fitting into a mini-mint tin.

#### 2.2.2 Rev A1

The Rev A1 of PocketBeagle was a prototype not released to production. A few lines were wrong to be able to control mikroBUS Click add-on board reset lines and they were adjusted.

#### 2.2.3 Rev A2

The Rev A2 of PocketBeagle was released to production and at World MakerFaire 2017.

Known issues in rev A2:

## **3.0 Connecting Up PocketBeagle**

This section provides instructions on how to hook up your board. The most common scenario is tethering PocketBeagle to your PC for local development.

## 3.1 What's In the Package

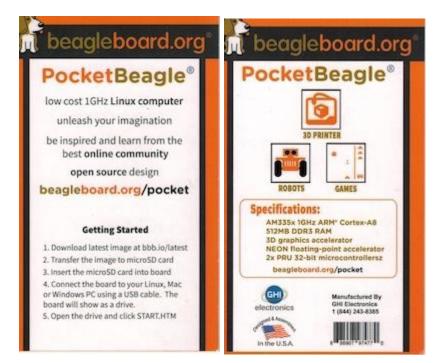
In the package you will find two items as shown in Figures 2 and 3.

- PocketBeagle
- Getting Started instruction card with link to the support URL.

Figure 2. PocketBeagle Package



Figure 3. PocketBeagle Package Insert



## 3.2 Connecting the board

This section will describe how to connect to the board. Information can also be found on the Quick Start Guide that came in the box. Detailed information is also available at <u>beagleboard.org/getting-started</u>

The board can be configured in several different ways, but we will discuss the most common scenario. Future revisions of this document may include additional configurations.

## 3.3 Tethered to a PC using Debian Images

In this configuration, you will need the following additional items:

- microUSB to USB Type A Cable
- microSD card (>=4GB and <128GB)

The board is powered by the PC via the USB cable, no other cables are required. The board is accessed either as a USB storage drive or via a web browser on the PC. You need to use either Firefox or Chrome on the PC, IE will not work properly. **Figure 4** shows this configuration.

**Figure 4. Tethered Configuration** 

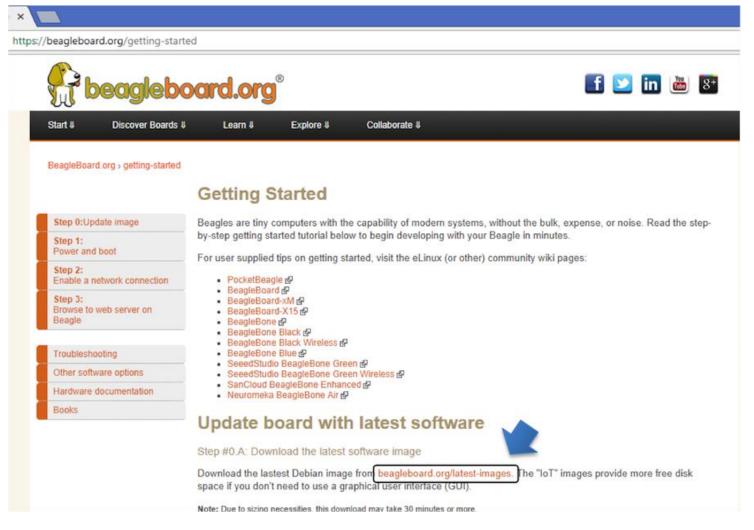


In some instances, such as when additional add-on boards, or PocketCapes are connected, the PC may not be able to supply sufficient power for the full system. In that case, review the power requirements for the add-on board/cape; additional power may need to be supplied via the 5v input, but rarely is this the case.

#### 3.3.1 Getting Started

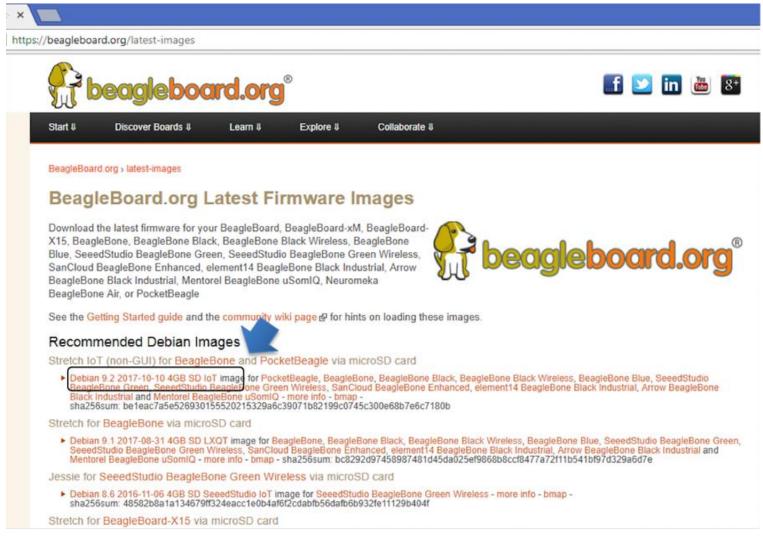
The following steps will guide you to quickly download a PocketBeagle software image onto your microSD card and get started writing code.

1. Navigate to the Getting Started Page <u>beagleboard.org/getting-started</u> Follow along with the instructions and click on the link noted in Figure 5 below <u>beagleboard.org/latest-images</u>. You can also get to this page directly by going to <u>bbb.io/latest</u>



2. Download the latest image onto your computer by following the link to the latest image and click on the Debian image for Stretch IoT (non-GUI) for BeagleBone and PocketBeagle via microSD card. See Figure 6 below. This will download a .img.xz file into the downloads folder of your computer.

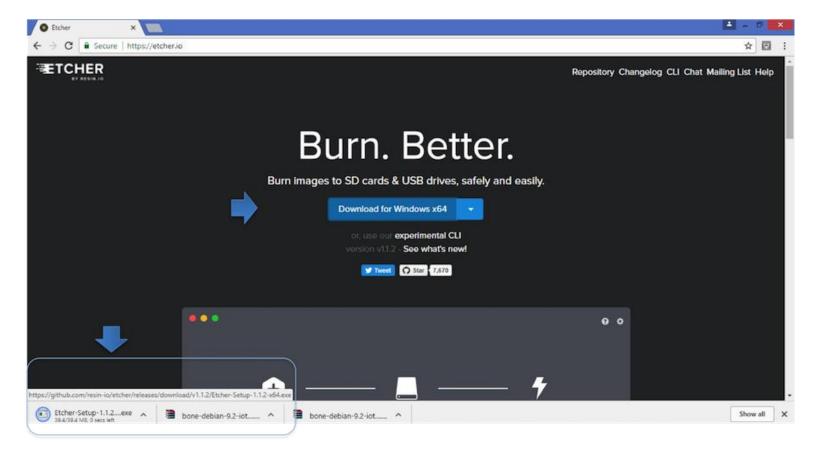
Figure 6. Download Latest Software Image



#### 3. Transfer the image to a microSD card.

Download and install an SD card programming utility if you do not already have one. We like <u>https://etcher.io/</u> for new users and so we show that one in the steps below. Go to your downloads folder and doubleclick on the .exe file and follow the on-screen prompts. See figure 7.

Figure 7. Download Etcher SD Card Utility

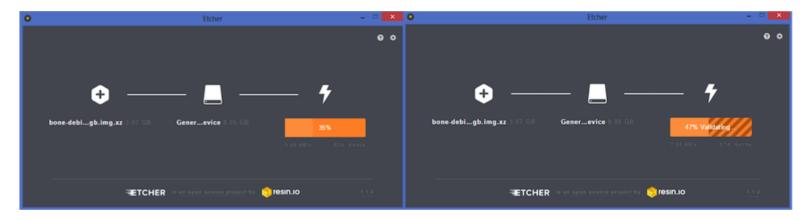


Insert a new microSD card into a card reader/writer and attach it via the USB connection to your computer. Follow the instructions on the screen for selecting the .img file and burning the image from your computer to the microSD card. Eject the SD card reader when prompted and remove the card. See Figures 8 and 9.

0	Etcher	<b>X</b>	0	Open		×
		00	🐑 🎯 🔹 🕆 🚺 🕨 Th	is PC + Downloads + v (	Search Downloa	ads 🔎
			Organize 👻 New folde	IF.		H • 🔲 0
			A 🔆 Favorites	Name	Date modified	Туре
			E Desktop	KeysightHandheldMeterLogger_3_1_5113	10/12/2017 10:53	File folder
			Downloads	MobaXterm_Installer_v10.4	10/30/2017 10:41	File folder
			Uropbox	bone-debian-9.2-iot-armhf-2017-10-10	11/2/2017 5:25 PM	WinRAR archive
			💯 Recent places	KeysightHandheldMeterLogger_3_1_5113	10/12/2017 10:48	Compressed (zipp
				MobaXterm_Installer_v10.4	10/30/2017 10:38	Compressed (zipp
Select image			D 🧥 OneDrive			
			🖻 🔩 Homegroup			
			🖌 🌉 This PC			
			🖻 🚂 Desktop			
			Documents			
ETCHER	is an open source project by 😒 👩	sin.io	🖻 🖬 Downloads 🛛 🎽	<		>
			File na	ime:	V OS Images	~
			107.655		0	Cancel
bone-debian-9.2-iot ^ 📓 bone-debian	-9.2-int				Open	Cancel

Figure 8. Select the PocketBeagle Image

Figure 9. Burn the Image to the SD Card



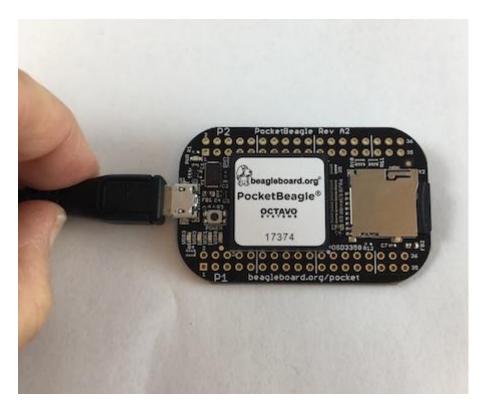
4. Insert the microSD card into the board - you'll hear a satisfying click when it seats properly into the slot. It is important that your microSD card is fully inserted prior to powering the system.

Figure 10. Insert the microSD Card into PocketBeagle



5. Connect the micro USB connector on your cable to the board as shown in Figure 11. The microUSB connector is fairly robust, but we suggest that you not use the cable as a leash for your PocketBeagle. Take proper care not to put too much stress on the connector or cable.

Figure 11. Insert the micro USB Connector into PocketBeagle



6. Connect the large connector of the USB cable to your Linux, Mac or Windows PC USB port as shown in Figure 12. The board will power on and the power LED will be on as shown in Figure 13 below.

Figure 12. Insert the USB connector into PC

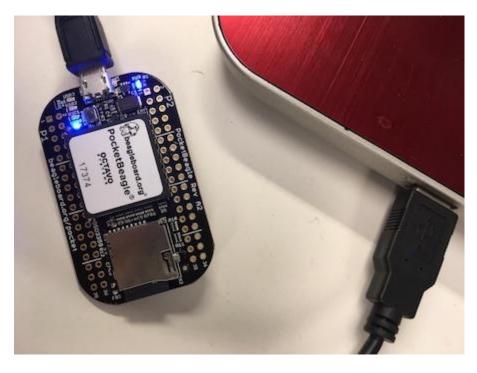
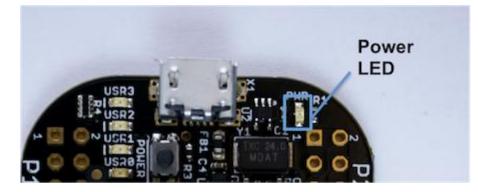


Figure 13. Board Power LED



7. As soon as you apply power, the board will begin the booting process and the userLEDs **Figure 14**will come on in sequence as shown below. It will take a few seconds for the status LEDs to come on, like teaching PocketBeagle to 'stay'. The LEDs will be flashing as it begins to boot the Linux kernel. While the four user LEDS can be over written and used as desired, they do have specific meanings in the image that you've initially placed on your microSD card once the Linux kernel has booted.

- **USER0** is the heartbeat indicator from the Linux kernel.
- **USER1** turns on when the microSD card is being accessed
- **USER2** is an activity indicator. It turns on when the kernel is not in the idle loop.
- USER3 idle

Figure 14. User LEDs



#### 3.3.2 Accessing the Board and Getting Started with Coding

The board will appear as a USB Storage drive on your PC after the kernel has booted, which will take approximately 10 seconds. The kernel on the board needs to boot before the port gets enumerated. Once the board appears as a storage drive, do the following:

1. Open the USB Drive folder to view the files on your PocketBeagle.

2. Launch Interactive Quick Start Guide.

Right Click on the file named **START.HTM** and open it in Chrome or Firefox. This will use your browser to open a file running on PocketBeagle via the microSD card. You will see **file:///Volumes/BEAGLEBONE/START.htm** in the url bar of the browser. See Figure 15 below. This action displays an interactive Quick Start Guide from PocketBeagle.

😚 Getting started with Beagle 🛛 🗙							
() file:///Volumes/BEAGLEBONE/START.h	itm						
	beaglebo	ard.org°	2		f 💟 in 💩 👫		
Star	rt 🥼 Discover Boards 🦆	Leam 🌡	Explore 🤳	Collaborate J	Google Custom Search Search		
			nputers with the ca		s, without the bulk, expense, or noise. Read the step-		
	ep 1: 3 ower and boot			o begin developing with you ed, visit the eLinux (or othe	an and an		
	ep 2: nable a network connection	BeagleBoard      BeagleBoard-xl	MeP				
Bri	ep 3: owse to web server on sagle	BeagleBoard-X15 g     BeagleBone Black g     BeagleBone Black g					
Tr	oubleshooting	<ul> <li>SeeedStudio Br</li> </ul>	ue di leagleBone Green d leagleBone Green V gleBone Enhanced	Wireless @			
Up	odate to latest software	Neuromeka Bea		L7			
Ot	her software options	If any step fails, it is recommended to update to the latest software image to use the instructions below.					
Ha	ardware documentation	Step 1 Power and boot					
Bo	ooks	step 1					
		to your computer. If y	you provide your on and the "type-A"	own, ensure it is of good qu plug to your computer. Not	o provide both power to your Beagle and connectivity ality. You'll connect the "type-B" plug d? of the USB e that BeagleBoard-X15 must always be powered		
		Alternatively, for Bear adapter connected to		eagleBoard-X15 and Beag	eBone Blue that require 12V, you can utilize a 5V		

3. Enable a Network Connection.

Click on 'Step 2' of the Interactive Quick Start Guide page to follow instructions to "Enable a Network Connection" (pointing to the DHCP server that is running on PocketBeagle). Copy the appropriate IP Address from the chart (according to your PC operating system type) and paste into your browser then add a **:3000** to the end of it. See example in Figure 16 below. This will launch from PocketBeagle one of it's favorite Web Based Development Environments, Cloud9 IDE, (Figure 17) so that you can teach your beagle new tricks!

Figure 16. Enable a Network Connection

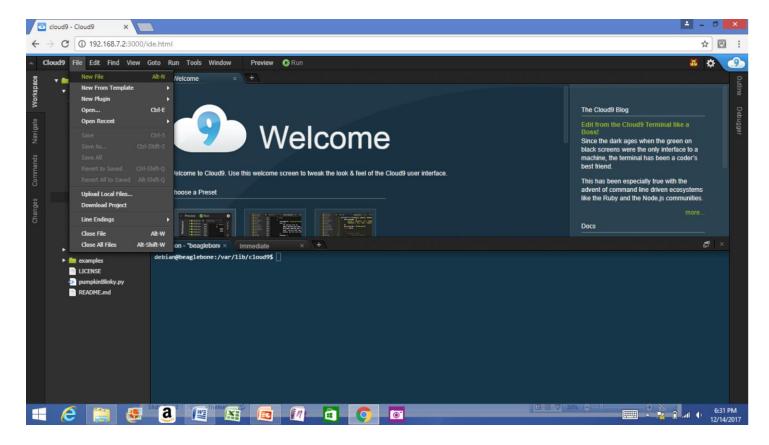
192.168.6.2:3000							
		Step 2	Enable a network	connection			
•		If connected via USB, a network adapter should show up on your computer. Your Beagle should be r server that will provide your computer with an IP address of either 192.168.7.1 or 192.168.6.1, deper USB network adapter supported by your computer's operating system. Your Beagle will reserve 192. 192.168.6.2 for itself.					
		If your Beagle includes WiFi, an access point called "BeagleBone-XXXX" where "XXXX" varies between boards. The access point password defaults to "BeagleBone". Your Beagle should be running a DHCP server that will provide your computer with an IP address in the 192.168.8.x range and reserve 192.168.8.1 for itself. If your Beagle is connected to your local area network (LAN) via either Ethernet or WiFi, it will utilize mDNS go to broadcast itself to your computer. If your computer supports mDNS, you should see your Beagle as beaglebone.local. Non-BeagleBone boards will utilize alternate names. Multiple BeagleBone boards on the same network will add a suffix					
	Step 1: O Power and boot						
	Step 2: Enable a network connection	such as beaglebon					
	Step 3: Browse to web server on Beagle			addresses and should out HTTPS security for the		o indicate an active connection. n to work.	
		IP Address	Connection Type	Operating System(s)	Status		
	Troubleshooting Update to latest software	192.168.7.2	USB	Windows	Inactive		
	Other software options	192.168.6.2	USB	Mac OS X, Linux	Active d		
	Hardware documentation	192.168.8.1	WIFI	all	Inactive		
	Books	beaglebone.local	all	mDNS enabled	Active P		
			1776				
		beaglebone-2.local	all	mDNS enabled	Inactive		

#### Figure 17. Launch Cloud9 IDE

👔 BeagleBoard.org - gettir 🗙 🖸 Cloud9 🗙 🛄	A -	0	×
← → C ① 192.168.7.2:3000/ide.html	Ŕ		9
Cloud9			
a Starke			

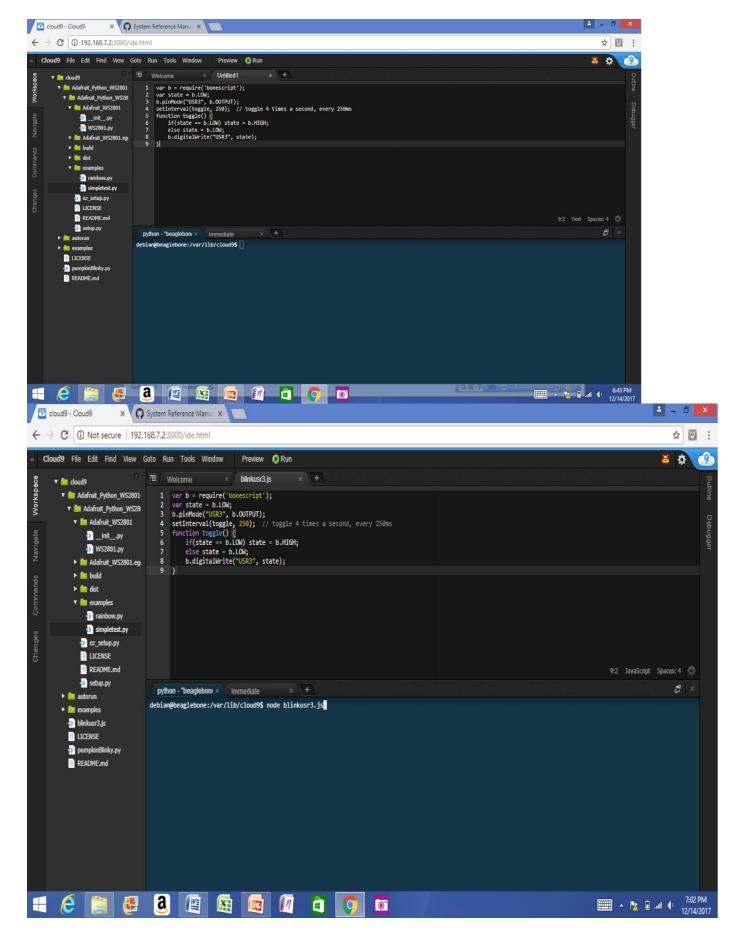
4. Get Started Coding with Cloud9 IDE - blinking USR3 LED in JavaScript using the BoneScript library example

1. Create a new text file



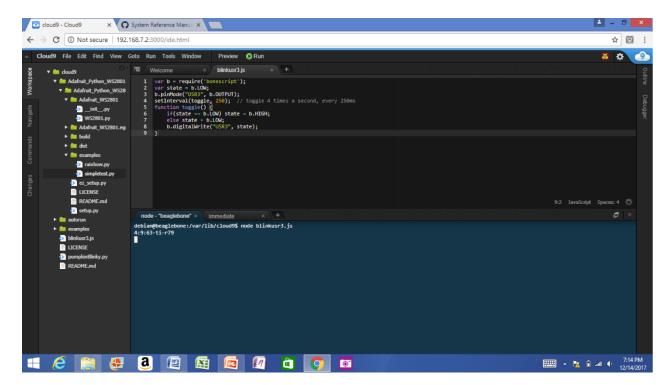
2. Copy and paste the below code into the editor

```
3. var b = require('bonescript');
4. var state = b.LOW;
5. b.pinMode("USR3", b.OUTPUT);
6. setInterval(toggle, 250); // toggle 4 times a second, every 250ms
7. function toggle() {
8. if(state == b.LOW) state = b.HIGH;
9. else state = b.LOW;
10. b.digitalWrite("USR3", state);
11.}
```



- 12. Save the new text file as *blinkusr3.js* within the default directory
- 13. Execute

within the default (/var/lib/cloud9) directory



14. Type CTRL+C to stop the program running

#### 3.3.3 Powering Down

1. Standard Power Down Press the power button momentarily with a tap. The system will power down automatically. This will shut down your software with grace. Software routines will run to completion.

The Standard Power Down can also be invoked from the Linux command shell via "sudo shutdown -h now".

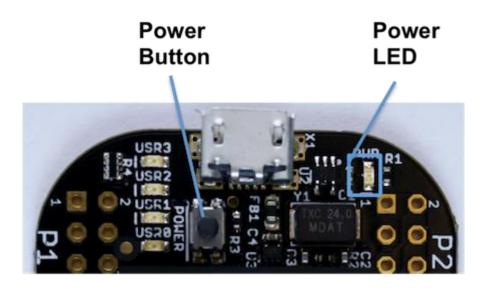
2. Hard Power Down Press the power button for 10 seconds. This will force an immediate shut down of the software. For example you may lose any items you have written to the memory. Holding the button longer than 10 seconds will perform a power reset and the system will power back on.

3. Remove the USB cable Remember to hold your board firmly at the USB connection while you remove the cable to prevent damage to the USB connector.

4. Powering up again. If you'd like to power up again without removing the USB cable follow these instructions:

- 1. If you used Step 1 above to power down, to power back up, hold the power button for 10 seconds, release then tap it once and the system will boot normally.
- 2. If you used Step 2 above to power down, to power back up, simply tap the power button and the system will boot normally.

Figure 20. Power Button



## 3.4 Other ways to Connect up to your PocketBeagle

The board can be configured in several different ways. Future revisions of this document may include additional configurations.

As other examples become documented, we'll update them on the Wiki for PocketBeaglegithub.com/beagleboard/pocketbeagle/wiki See also the on-line discussion.

## 4.0 PocketBeagle Overview

PocketBeagle is built around Octavo Systems' OSD335x-SM System-In-Package that integrates a highperformance Texas Instruments AM3358 processor, 512MB of DDR3, power management, nonvolatile serial memory and over 100 passive components into a single package. This integration saves board space by eliminating several packages that would otherwise need to be placed on the board, but more notably simplifies our board design so we can focus on the user experience.

The compact PocketBeagle design also offers access through the expansion headers to many of the interfaces and allows for the use of add-on boards called PocketCapes and Click Boards from MikroElektronika, to add many different combinations of features. A user may also develop their own board or add their own circuitry.

## 4.1 PocketBeagle Features and Specification

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

#### Table 3. PocketBeagle Features

Feature	,
System-In-	Octavo Systems OSD335x-SM in 256 Ball BGA (21mm x 21mm)

Package	
SiP Incorporates :	
Processor	Texas Instruments 1GHz Sitara <sup>™</sup> AM3358 ARM® Cortex®-A8 with NEON floating-point accelerator
Graphics Engine	Imagination Technologies PowerVR SGX530 Graphics Accelerator
Real-Time Units	2x programmable real-time unit (PRU) 32-bit 200MHz microcontrollers with single-cycle I/O latency
Coprocessor	ARM® Cortex®-M3 for power management functions
SDRAM Memory	512MB DDR3 800MHz RAM
Non-Volatile Memory	4KB I2C EEPROM for board configuration information
Power Management	TPS65217C PMIC along with TL5209 LDO to provide power to the system with integrated 1-cell LiPo battery support
Connectivity :	
SD/MMC	Bootable microSD card slot
USB	High speed USB 2.0 OTG (host/client) micro-B connector
Debug Support	JTAG test points and gdb/other monitor-mode debug possible
Power Source	microUSB connector, also expansion header options (battery, VIN or USB-VIN)
User I/O	Power Button with press detection interrupt via TPS65217C PMIC
Expansion Header :	
USB	High speed USB 2.0 OTG (host/client) control signals
Analog Inputs	8 analog inputs with 6 @ 1.8V and 2 @ 3.3V along with 1.8V references
Digital I/O	44 digital GPIOs accessible with 18 enabled by default including 2 shared with the 3.3V analog input pins

UART	3 UARTs accessible with 2 enabled by default
I2C	2 I2C busses enabled by default
SPI	2 SPI busses with single chip selects enabled by default
PWM	4 Pulse Width Modulation outputs accessible with 2 enabled by default
QEP	2 Quadrature encoder inputs accessible
CAN	2 CAN bus controllers accessible

#### 4.1.1 OSD3358-512M-BSM System in Package

The Octavo Systems OSD3358-512M-BSM System-In-Package (SiP) is part of a family of products that are building blocks designed to allow easy and cost-effective implementation of systems based in Texas Instruments powerful Sitara AM335x line of processors. The OSD335x-SM integrates the AM335x along with the TI TPS65217C PMIC, the TI TL5209 LDO, up to 1 GB of DDR3 Memory, a 4 KB EEPROM for non-volatile configuration storage and resistors, capacitors and inductors into a single 21mm x 21mm design-in-ready package.

With this level of integration, the OSD335x-SM family of SiPs allows designers to focus on the key aspects of their system without spending time on the complicated high-speed design of the processor/DDR3 interface or the PMIC power distribution. It reduces size and complexity of design.

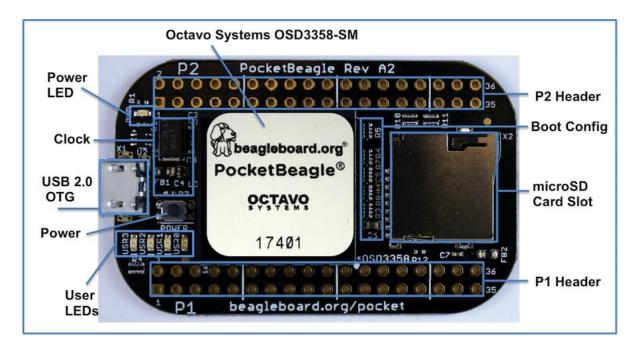
Full Datasheet and more information is available at <u>octavosystems.com/octavo\_products/osd335x-sm/</u>

## 4.2 Board Component Locations

This section describes the key components on the board, their location and function.

Figure 21 below shows the locations of the devices, connectors, LEDs, and switches on the PCB layout of the board.

Figure 21. Key Board Component Locations



#### **Key Components**

- The Octavo Systems OSD3358-512M-BSM System-In-Package is the processor system for the board
- P1 and P2 Headers come unpopulated so a user may choose their orientation
- User LEDs provides 4 programmable blue LEDs
- **Power BUTTON** can be used to power up or power down the board (see section 3.3.3 for details)
- **USB 2.0 OTG** is a microUSB connection to a PC that can also power the board
- **Power LED** provides communication regarding the power to the board
- **microSD** slot is where a microSD card can be installed.

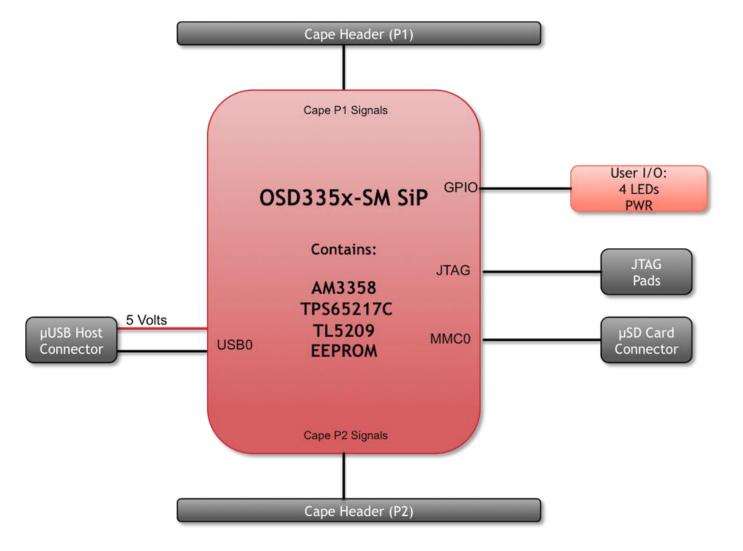
## **5.0 PocketBeagle High Level Specification**

This section provides the high level specification of PocketBeagle.

### 5.1 Block Diagram

Figure 22 below is the high level block diagram of PocketBeagle.

Figure 22. PocketBeagle Key Components



## 5.2 System in Package (SiP)

The OSD335x-SM Block Diagram is detailed in Figure 23 below. More information, including design resources are available on the <u>'Octavo Systems Website'</u>

Figure 23. OSD335x SIP Block Diagram

Octavo Systems OSD335x-SM 256 Ball BGA (21mm x 21mm)							
TPS65217C Power In 5V: • DC, USB, Li-ion Battery Power Out: • 1.8V, 3.3V, SYS TL5209 Power Out: • 3.3V	TI AM335x ARM® Cortex®-A8 • Up to 1 GHz clock • 32KB L1 Icache + SED • 32KB L1 Dcache + SED • 256KB L2 cache + ECC • 64KB dedicated RAM • 64KB shared L3 RAM • 64KB shared L3 RAM	System • ADC (8 channel) 12-bit SAR • PRU-ICSS (PRU x2) • RTC • Timers x8 • eHRPWM x3 • eQEP x3 • eCAP x3 • Crystal oscillator x2 • JTAG					
main memory 4KB EEPROM Passive Components	Serial • UART x6, SPI x2, I2C x3 • McASP x2 (4 channel) • CAN x2 (Ver 2A and B) • USB2.0 HS OTG+PHY x2 • Ethernet 10/100/1000 2- port and switch	LCD Display • Up to 24-bit color • 3D Graphics Engine • Character Display • Active Matrix LCD • Passive Matrix LCD • Touch screen					

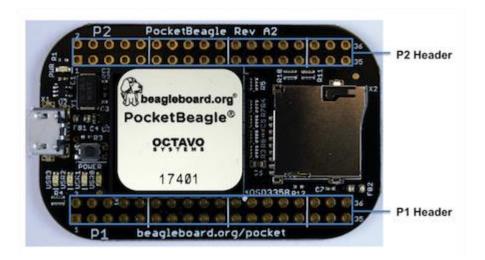
Note: PocketBeagle utilizes the 512MB DDR3 memory size version of the OSD335x-SM A few of the features of the OSD335x-SM SiP may not be available on PocketBeagle headers. Please check Section 7 for the P1 and P2 header pin tables.

## 5.3 Connectivity

### 5.3.1 Expansion Headers

PocketBeagle gives access to a large number of peripheral functions and GPIO via 2 dual rail expansion headers. With 36 pins each, the headers have been left unpopulated to enable users to choose the header connector orientation or add-on board / cape connector style. Pins are clearly marked on the bottom of the board with additional pin configurations available through software settings. Detailed information is available in Section 7.

Figure 24. PocketBeagle Expansion Headers

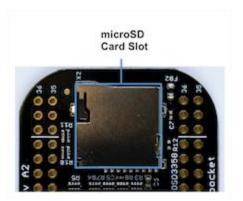


#### 5.3.2 microSD Connector

The board is equipped with a single microSD connector to act as the primary boot source for the board. Just about any microSD card you have will work, we commonly find 4G to be suitable.

When plugging in the SD card, the writing on the card should be up. Align the card with the connector and push to insert. Then release. There should be a click and the card will start to eject slightly, but it then should latch into the connector. To eject the card, push the SD card in and then remove your finger. The SD card will be ejected from the connector. Do not pull the SD card out or you could damage the connector.

Figure 25. microSD Connector



#### 5.3.3 USB 2.0 Connector

The board has a microUSB connector that is USB 2.0 HS compatible that connects the USB0 port to the SiP. Generally this port is used as a client USB port connected to a power source, such as your PC, to power the board. If you would like to use this port in host mode you will need to supply power for peripherals via Header P1 pin 7 (USB1.VIN) or through a powered USB Hub. Additionally, in the USB host configuration, you will need to power the board through Header P1 pin 1 (VIN) or Header P1 pin 7 (USB1.VIN) or Header P2 pin 14 (BAT.VIN)

Figure 26. USB 2.0 Connector



#### 5.3.4 Boot Modes

There are three boot modes:

- **SD Boot**: MicroSD connector acts as the primary boot source for the board. This is described in Section 3.
- **USB Boot**: This mode supports booting over the USB port. More information can be found in the project called "BeagleBoot" This project ported the BeagleBone bootloader server BBBIfs(currently written in c) to JavaScript(node.js) and make a cross platform GUI (using electron framework) flashing tool utilizing the etcher.io project. This will allow a single code base for a cross platform tool. For more information on BeagleBoot, see the <u>BeagleBoot Project Page</u>.
- Serial Boot: This mode will use the serial port to allow downloading of the software. A separate USB to TTL level <u>serial UART converter cable</u> is required or you can connect one of the Mikroelektronika <u>FTDI Click Boards</u> to use this method. The UART pins on PocketBeagle's expansion headers support the interface. For more information regarding the pins on the expansion headers and various modes, see Section 7.

Header.Pin	Silkscreen	Proc Ball	SiP Ball	Pin Name (Mode 0)
P1.22	GND			GND
P1.30	U0_TX	E16	B12	uart0_txd
P1.32	U0_RX	E15	A12	uart0_rxd

Table 4. UART Pins on Expansion Headers for Serial Boot

If the Serial Boot is not in use, the UARTO pins can be used for Serial Debug. See Section 5.6 for more information.

# Software to support USB and serial boot modes is not provided by beagleboard.org. Please contact TI for support of this feature.

### 5.4 Power

The board can be powered from three different sources:

- A USB port on a PC.
- A power supply with a USB connector.

• Expansion Header pins.

**Note:** VIN-USB is directly shorted between the USB connector on PocketBeagle and USB1\_VI on the expansion headers. You should only source power to the board over one of these and may optionally use the other as a power sink.

The tables below show the power related pins available on PocketBeagle's Expansion Headers.

Header.Pin	Silkscreen	Proc Ball	SiP Ball	Pin Name (Mode 0)
P1.01	VIN		P10, R10, T10	VIN
P1.07	USB1_VI		P9, R9, T9	VIN-USB
P2.14	BAT_+		P8, R8, T8	VIN-BAT

Table 5. Power Inputs Available on Expansion Headers

Table 6. Power Outputs Available on Expansion Headers

Header.Pin	Silkscreen	Proc Ball	SiP Ball	Pin Name (Mode 0)
P1.14	+3.3V		F6, F7, G6, G7	VOUT-3.3V
P1.24	VOUT		K6, K7, L6, L7	VOUT-5V
P2.13	VOUT		K6, K7, L6, L7	VOUT-5V
P2.23	+3.3V		F6, F7, G6, G7	VOUT-3.3V

Table 5. Ground Pins Av	ilable on Expansion Headers
-------------------------	-----------------------------

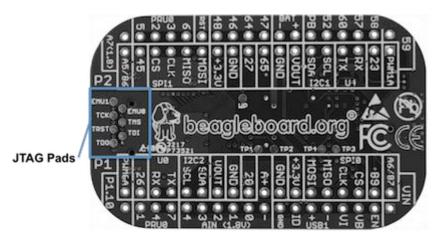
Header.Pin	Silkscreen	Proc Ball	SiP Ball	Pin Name (Mode 0)
P1.15	USB1_GND			GND
P1.16	GND			GND
P1.22	GND			GND
P2.15	GND			GND
P2.21	GND			GND

**Note**: A comprehensive tutorial for Power Inputs and Outputs for the OSD335x System in Package is available in the <u>'Tutorial Series'</u> on the Octavo Systems website.

## 5.5 JTAG Pads

Pads for an optional connection to a JTAG emulator has been provided on the back of PocketBeagle. More information about JTAG emulation can be found on the TI website - <u>'Entry-level debug through full-</u> <u>capability development'</u>

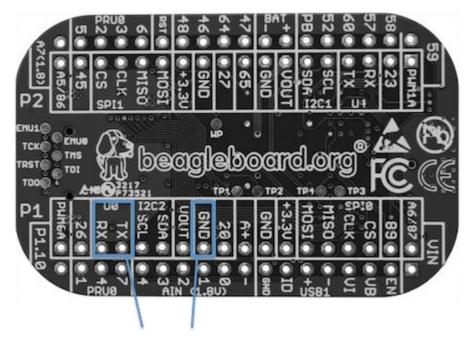
Figure 27. JTAG Pad Connections



## 5.6 Serial Debug Port

Serial debug is provided via UART0 on the processor. See Section 5.3.4 for the Header Pin table. Signals supported are TX and RX. None of the handshake signals (CTS/RTS) are supported. A separate USB to TTL level <u>serial UART converter cable</u> is required or you can connect one of the Mikroelektronika <u>FTDI Click</u> <u>Boards</u>to use this method.

Figure 28. Serial Debug Connections



**Serial Debug Connections** 

If serial boot is not used, the UARTO can be used to view boot messages during startup and can provide access to a console using a terminal access program like <u>Putty</u>. To view the boot messages or use the console the UART should be set to a baud rate of 115200 and use 8 bits for data, no parity bit and 1 stop bit (8N1).

## 6.0 Detailed Hardware Design

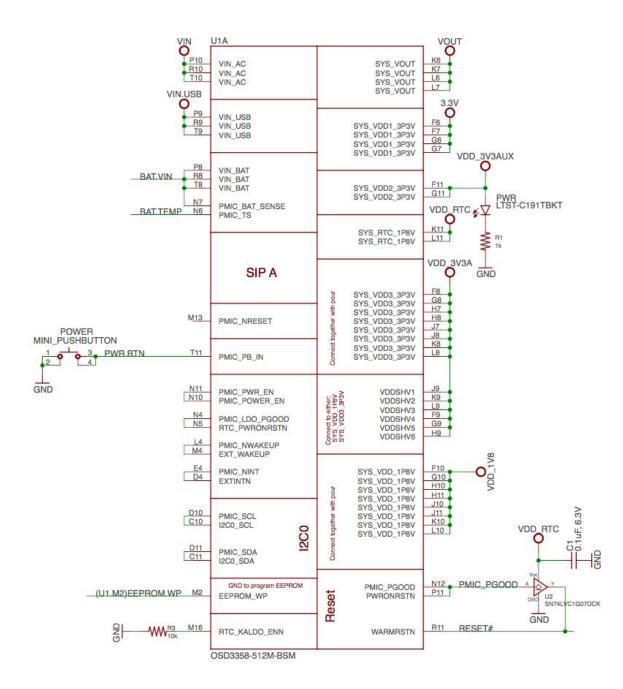
The following sections contain schematic references for PocketBeagle. Full schematics in both PDF and Eagle are available on the <u>'PocketBeagle Wiki'</u>

## 6.1 OSD3358-SM SiP Design

Schematics for the OSD3358-SM SiP are divided into several diagrams.

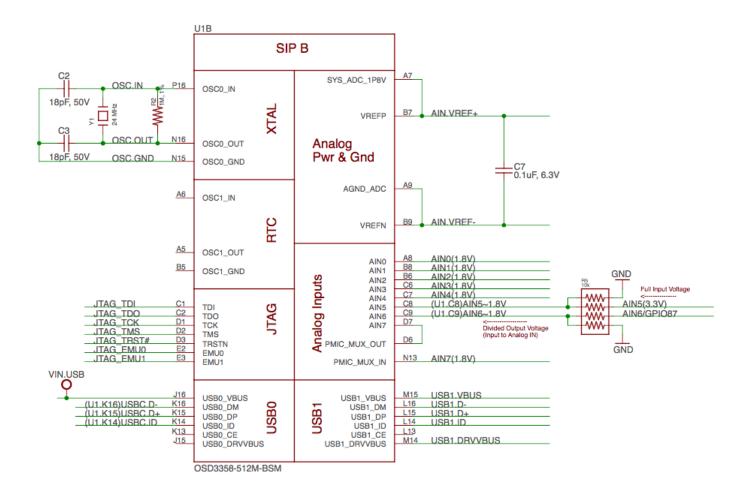
#### 6.1.1 SiP A OSD3358 SiP System and Power Signals

Figure 29. SiP A OSD3358 SiP System and Power Signals



#### 6.1.2 SiP B OSD3358 SiP JTAG, USB & Analog Signals

Figure 30. SiP B OSD3358 SiP JTAG, USB & Analog Signals



6.1.3 SiP C OSD3358 SiP Peripheral Signals

Figure 31. SiP C OSD3358 SiP Peripheral Signals

		U1C					
UART0.TX UART0.RX	B12 A12	UART0_TXD UART0_RXD	SIP C	GPMC_CLK		GPIO65	_
SPI1.MISO	C12	UARTO_CTSN		GPMC_A0	T12		
SPI1.MOSI	C13	UARTO_RTSN		GPMC_A1	R12		
				GPMC_A2	P12 T13	PWM1A	
I2C1.SCL	B11			GPMC_A3	R13	GPIO52	
12C1.SDA	A11	UART1_TXD		GPMC_A4	P13		
I2C2.SDA	B10	UART1_RXD UART1_CTSN		GPMC_A5 GPMC_A6	T14		
I2C2.SCL	A10	UART1_RTSN		GPMC_A8 GPMC_A7	R14	(U1.R14)USR2	
		over 1 There		GPMC_A8	P14		
				GPMC_A9	T15		
SPI0.CLK SPI0.MISO	A13 B13	SPI0_SCLK		GPMC_A10	R15 T16		_
SPI0.MISO SPI0.MOSI	B13	SPI0_D0		GPMC_A11	110	GPIO59	_
SPI0.CS	A14	SPI0_D1			N14	GPIO60	
(U1.C14)MMC0.CD	C14	SPI0_CS0 SPI0_CS1		GPMC_BEN1 GPMC_WAIT0	P15		
(01.01.)		580_051		GPMC_WATTO GPMC_WPN	R16		
U1.B15)MMC0.CLK	B15	MMC0_CLK		GPMC CSN0	P3 P2		
(U1.B16)MMC0.CMD	B16	MMC0_CMD		GPMC_CSN1	P2		
(U1.A16)MMC0.D0 (U1.A15)MMC0.D1	A16 A15	MMC0_DAT0		GPMC_CSN2	P1 R7	GPIO64	
(U1.C16)MMC0.D2	C16	MMC0_DAT1		GPMC_CSN3	<u> </u>	GF1064	
(U1.C15)MMC0.D2	C15	MMC0_DAT2		00110 100	R3		
(01.013)141400.03	0.0	MMC0_DAT3		GPMC_AD0	R2		
				GPMC_AD1 GPMC_AD2	R1		
PWM0A	A1	MCASP0_ACLKX		GPMC_AD2 GPMC_AD3	T3		
PRU0.1	A2	MCASP0_FSX		GPMC_AD3	T2		
PRU0.2	B2	MCASP0_AXR0		GPMC_AD5	T1		
PRU0.3	B1	MCASP0_AHCLKR		GPMC_AD6	P4		
PRU0.4	A3 B3	MCASP0_ACLKR		GPMC_AD7	R4		
PRU0.5 PRU0.6	C3	MCASP0_FSR			Т4		
	C4	MCASP0_AXR1		GPMC_AD8	P5	GPIO23	
		MCASP0_AHCLKX		GPMC_AD9	R5	GPIO26	
				GPMC_AD10 GPMC_AD11	T5	GPIO27	
SPI1.CS	A4	XDMA_EVENT_INTR0		GPMC_AD12	P6	GPIO48	
GPIO20	B4	XDMA_EVENT_INTR1		GPMC_AD13	R6	GPIO45	
SPI1.CLK	<u>C5</u>	ECAP0_IN_PWM0_OUT		GPMC_AD14	T6	GPIO46	
				GPMC_AD15	P7	GPIO47	
					M1		
				GPMC_ADVN_ALE	N3	(U1.M2)EEPROM.WP	WP
				GPMC_BEN0_CLE	N2	(	
				GPMC_WEN GPMC_OEN_REN	N1		
				GI MO_OCH_HEN			
		OSD3358-512M-BSM					

### 6.1.4 SiP D OSD3358 SiP System Boot Configuration

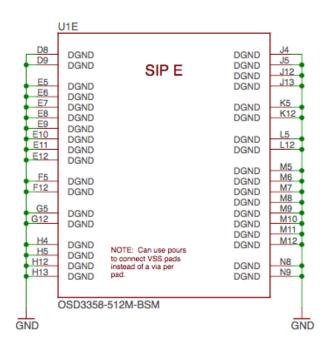
Figure 32. SiP D OSD3358 SiP System Boot Configuration

H16       MII1_TX_CLK       LCD_DATA0       G3_(U1.G3)LCD_D0.B3         H15       MII1_TXD0       LCD_DATA1       G1_(L1.G2)LCD_D1.B4         H14       MII1_TXD1       LCD_DATA2       G1_(L1.G2)LCD_D2.B5         G15       MII1_TXD2       LCD_DATA3       H3_(U1.H3)LCD_D3.B6         G14       MII1_TX_EN       LCD_DATA4       H1_(U1.H1)LCD_D5.G2         F14       MII1_CCL       LCD_DATA6       J3_(U1.J3)LCD_D6.G3         F14       MII1_CCL       LCD_DATA6       J3_(U1.J2)LCD_D7.G4         MI1_COL       LCD_DATA6       J3_(U1.L3)LCD_D9.G6         K11_UL_X1_L       LCD_DATA10       LCD_DATA10         MI1_RXD0       LCD_DATA10       K1_(U1.K1)LCD_D11.B3         MI1_RXD1       LCD_DATA11       K1_(U1.K1)LCD_D11.B3         MI1_RXD3       LCD_DATA11       K1_(U1.K1)LCD_D11.B3         MI1_RXD3       LCD_DATA11       K1_(U1.L3)LCD_D12.B4         MI1_RX_D7       LCD_DATA11       K1_(U1.L3)LCD_D14.B6         MI1_RX_D7       LCD_DATA14       M3_(U1.M3)LCD_D15.B7         J14       MI1_REF_CLK       LCD_PCLK         MI1_RX_D7       LCD_AC_BIAS_EN       F1_PRU1.10         F3_AIN5(3.3V)       F2_AIN6/GPI087         E1_GPI089       E1_GPI089 </th <th></th> <th>U1D</th> <th></th> <th></th>		U1D		
OSD3358-512M-BSM	H15 H14 G16 G15 G14 F14 F15 E16 D15 D14 E15 F16 J14 D13 E13	MII1_TXD0 MII1_TXD1 MII1_TXD2 MII1_TXD3 MII1_TX_EN MII1_CRS MII1_CCL MII1_RX_CLK MII1_RXD0 MII1_RXD1 MII1_RXD1 MII1_RXD2 MII1_RXD3 MII1_RX_ER MII1_RX_ER MII1_RX_DV RMII1_REF_CLK MDC MDIO	LCD_DATA1 LCD_DATA2 LCD_DATA3 LCD_DATA4 LCD_DATA5 LCD_DATA6 LCD_DATA6 LCD_DATA7 LCD_DATA9 LCD_DATA10 LCD_DATA10 LCD_DATA12 LCD_DATA12 LCD_DATA13 LCD_DATA13 LCD_DATA15 LCD_PCLK LCD_VSYNC LCD_HSYNC	G2 (U1.G2)LCD.D1.B4 G1 (U1.G1)LCD.D2.B5 H3 (U1.H3)LCD.D3.B6 H2 (U1.H2)LCD.D4.B7 H1 (U1.H1)LCD.D5.G2 J3 (U1.J3)LCD.D6.G3 J2 (U1.J2)LCD.D7.G4 J1 (U1.J1)LCD.D8.G5 K3 (U1.K3)LCD.D9.G6 K2 (U1.K2)LCD.D10.G7 K1 (U1.K1)LCD.D10.G7 K1 (U1.K1)LCD.D12.R4 L2 (U1.L2)LCD.D12.R4 L2 (U1.L2)LCD.D13.B5 L1 (U1.L1)LCD.D14.R6 M3 (U1.M3)LCD.D15.R7 F1 PRU1.10 F3 AIN5(3.3V) F2 AIN6/GPI087

#### 6.1.5 SiP E OSD3358 SiP Power Signals

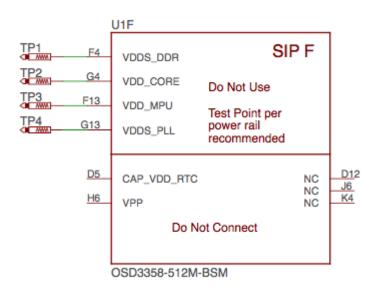
LIAD

Figure 33. SiP E OSD3358 SiP Power Signals



#### 6.1.6 SiP F OSD3358 SiP Power Signals

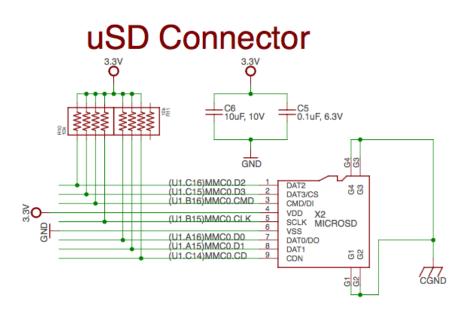
Figure 34. SiP F OSD3358 SiP Power Signals



### **6.2 MicroSD Connection**

The Micro Secure Digital (microSD) connector design is highlighted in Figure 35.

Figure 35. microSD Connections



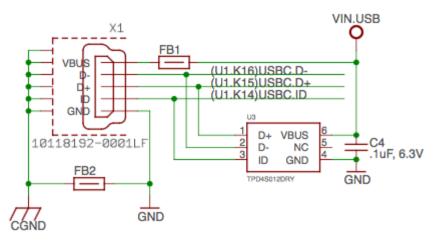
## 6.3 USB Connector

The USB connector design is highlighted in Figure 36.

Note that there is an ID pin for dual-role (host/client) functionality. The hardware fully supports it, but care should be taken to ensure the kernel in use is either statically or dynamically configured to recognize and utilize the proper mode.

Figure 36. USB Connection

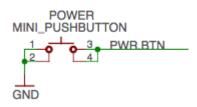
# **USB** Device



### 6.4 Power Button Design

The power button design is highlighted in Figure 37.

Figure 37. Power Button



### 6.5 User LEDs

There are four user programmable LEDs on PocketBeagle. The design is highlighted in Figure 38. Table 6 Provides the LED control signals and pins. A logic level of "1" will cause the LEDs to turn on.

Figure 38. User LEDs

# **USER LEDs**

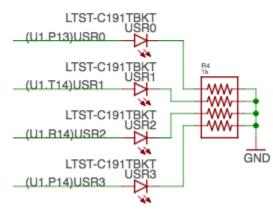


Table 6. User LED Control Signals/Pins

LED	Signal Name	Proc Ball	SiP Ball
USR0	GPIO1_21	V15	P13
USR1	GPIO1_22	U15	T14
USR2	GPIO1_23	T15	R14
USR3	GPIO1_24	V16	P14

## 6.6 JTAG Pads

There are 7 pads on the bottom of PocketBeagle to connect JTAG for debugging. The design is highlighted in Figure 39. More information regarding JTAG debugging can be found at <u>'www.ti.com/jtag'</u>

Figure 39. JTAG Pads Design

# JTAG Pads

_JTAG_TMS	
JTAG_TDI	
JTAG_TDO	
JTAG_TCK	
JTAG_EMU0	
JTAG_TRST# 10k	
_JTAG_EMU1	10

## 6.7 PRU-ICSS

The Programmable Real-Time Unit Subsystem and Industrial Communication SubSystem (PRU-ICSS) module is located inside the AM3358 processor, which is inside the Octavo Systems SiP. Commonly referred to as just the "PRU", this little subsystem will unleash a lot of performance for you to use in your application. Consisting of dual 32-bit RISC cores (Programmable Real-Time Units, or PRUs), data and instruction memories, internal peripheral modules, and an interrupt controller (INTC). The programmable nature of the PRU-ICSS, along with their access to pins, events and all SoC resources, provides flexibility in implementing fast real-time responses, specialized data handling operations, custom peripheral interfaces, and in offloading tasks from the other processor cores of the system-on-chip (SoC). Access to these pins is provided by PocketBeagle's expansion headers and is multiplexed with other functions on the board. Access is not provided to all of the available pins.

Some getting started information can be found on https://beagleboard.org/pru.

Additional documentation is located on the Texas Instruments website at *processors.wiki.ti.com/index.php/PRU-ICSS* and also located at *http://github.com/beagleboard/am335s\_pru\_package.* 

Example projects using the PRU-ICSS can be found at processors.wiki.ti.com/index.php/PRU Projects.

#### 6.7.1 PRU-ICSS Features

The features of the PRU-ICSS include:

- 32-Bit Load/Store RISC architecture
- 8K Byte instruction RAM (2K instructions) per core
- 8K Bytes data RAM per core
- 12K Bytes shared RAM
- Operating frequency of 200 MHz
- PRU operation is little endian similar to ARM processor
- All memories within PRU-ICSS support parity
- Includes Interrupt Controller for system event handling
- Fast I/O interface

- 16 input pins and 16 output pins per PRU core. (Not all of these are accessible on the PocketBeagle. Please check the Pin Table below for PRU-ICSS features available through the P1 and P2 headers.)

#### 6.7.2 PRU-ICSS Block Diagram

Figure 40 is a high level block diagram of the PRU-ICSS.

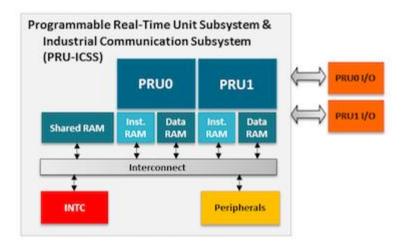


Figure 40. PRU-ICSS Block Diagram

#### 6.7.3 PRU-ICSS Pin Access

Both PRU 0 and PRU1 are accessible from the expansion headers. Listed below are the ports that can be accessed on each PRU.

Table 6. below shows which PRU-ICSS signals can be accessed on PocketBeagle and on which connector and pins on which they are accessible. Some signals are accessible on the same pins.

Table 6. PRU0 and PRU1 Access

Use scroll bar at bottom of chart to see additional features in columns to the right. When printing this document, you will need to print this chart separately.

Header. Pin	Silkscree n	Proces sor Ball	SiP Ba ll	Mode3	Mode4	Mode5	Mode6	Note
P1.02	A6/87	R5	F2			pr1_pru1_pru_r30_9 (Output)	pr1_pru1_pru_r 31_9 (Input)	
P1.04	89	R6	E1			pr1_pru1_pru_r30_11 (Output)	pr1_pru1_pru_r 31_11 (Input)	
P1.06	SPIO_CS	A16	A1 4		pr1_uart0_t xd (Output)			UAR T Tran smit Data
P1.08	SPIO_CL K	A17	A1 3		pr1_uart0_c ts_n (Input)			UAR T Clear to Send
P1.10	SPIO_MI SO	B17	B1 3		pr1_uart0_r ts_n (Output)			UAR T Requ est to Send
P1.12	SPIO_MO SI	B16	B1 4		pr1_uart0_r xd (Input)			UAR T Recei ve Data
P1.20	20	D14	B4			pr1_pru0_pru_r31_16 (Input)		
P1.26	I2C2_SD A	D18	B1 0			pr1_uart0_cts_n (Input)		UAR T Clear to Send
P1.28	I2C2_SC L	D17	A1 0			pr1_uart0_rts_n (Output)		UAR T Requ est to Send
P1.29	PRU0_7	A14	C4			pr1_pru0_pru_r30_7 (Output)	pr1_pru0_pru_r 31_7 (Input)	
P1.30	U0_TX	E16	B1 2			pr1_pru1_pru_r30_15 (Output)	pr1_pru1_pru_r 31_15 (Input)	
P1.31	PRU0_4	B12	A3			pr1_pru0_pru_r30_4 (Output)	pr1_pru0_pru_r 31_4 (Input)	
P1.32	U0_RX	E15	A1 2			pr1_pru1_pru_r30_14 (Output)	pr1_pru1_pru_r 31_14 (Input)	

P1.33	PRU0_1	B13	A2		pr1_pru0_pru_r30_1 (Output)	pr1_pru0_pru_r 31_1 (Input)	
P1.35	P1.10	V5	F1		pr1_pru1_pru_r30_10 (Output)	pr1_pru1_pru_r 31_10 (Input)	
P1.36	PWM0A	A13	A1		pr1_pru0_pru_r30_0 (Output)	pr1_pru0_pru_r 31_0 (Input)	
P2.09	I2C1_SC L	D15	B1 1		pr1_uart0_txd (Output)	pr1_pru0_pru_r 31_16 (Input)	UAR T Tran smit Data
P2.11	I2C1_SD A	D16	A1 1		pr1_uart0_rxd (Input)	pr1_pru1_pru_r 31_16 (Input)	UAR T Recei ve Data
P2.17	65	V12	Τ7		pr1_mdio_mdclk		MDIO Clk
P2.18	47	U13	Ρ7		pr1_ecap0_ecap_capin_a pwm_o	pr1_pru0_pru_r 31_15 (Input)	Enha nced captu re input or Auxil iary PWM out
P2.20	64	T13	R7		pr1_mdio_data		MDIO Data
P2.22	46	V13	Т6			pr1_pru0_pru_r 31_14 (Input)	
P2.24	48	T12	P6			pr1_pru0_pru_r 30_14 (Output)	
P2.28	PRU0_6	D13	C3		pr1_pru0_pru_r30_6 Output)	pr1_pru0_pru_r 31_6 (Input)	
P2.29	SPI1_CL K	C18	C5	pr1_ecap0_ecap_capin_a pwm_o			Enha nced captu re input or Auxil iary PWM out
P2.30	PRU0_3	C12	B1		pr1_pru0_pru_r30_3 (Output)	pr1_pru0_pru_r 31_3 (Input)	
P2.31	SPI1_CS	A15	A4		pr1_pru1_pru_r31_16		

				(Input)	
P2.32	PRU0_2	D12	B2	pr1_pru0_pru_r30_2 (Output)	pr1_pru0_pru_r 31_2 (Input)
P2.33	45	R12	R6		pr1_pru0_pru_r 30_15 (Output)
P2.34	PRU0_5	C13	В3	pr1_pru0_pru_r30_5 (Output)	pr1_pru0_pru_r 31_5 (Input)
P2.35	A5/86	U5	F3	pr1_pru1_pru_r30_8 (Output)	pr1_pru1_pru_r 31_8 (Input)

## 7.0 Connectors

This section describes each of the connectors on the board.

## 7.1 Expansion Header Connectors

The expansion interface on the board is comprised of two 36 pin connectors. The two Expansion Header Connectors on PocketBeagle are labeled P1 and P2. The connections are a standard 100 mil distance so that they can be compatible with many standard expansion items. The silkscreen for the headers on the bottom of the board provides the easiest way to identify them. See Figure 41.

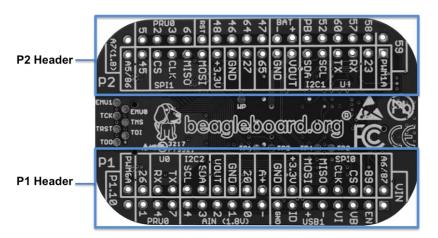


Figure 41. Expansion Headers for PocketBeagle

All signals on the expansion headers are **3.3V** unless otherwise indicated.

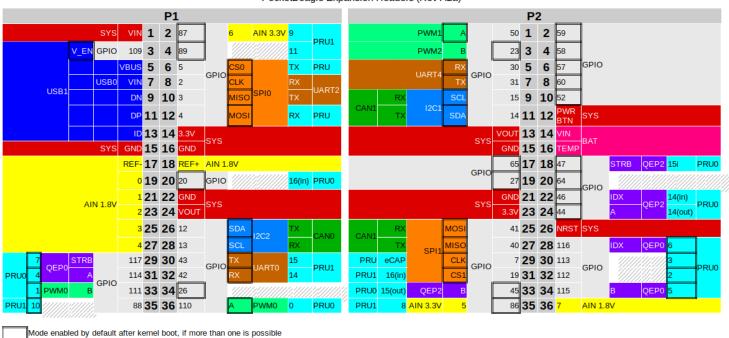
#### NOTE: Do not connect 5V logic level signals to these pins or the board will be damaged.

## NOTE: DO NOT APPLY VOLTAGE TO ANY I/O PIN WHEN POWER IS NOT SUPPLIED TO THE BOARD. IT WILL DAMAGE THE PROCESSOR AND VOID THE WARRANTY.

NO PINS ARE TO BE DRIVEN UNTIL AFTER THE SYS\_RESET LINE GOES HIGH.

Figure 42 shows a color coded chart with an overview of the most popular functions of PocketBeagle's Expansion Header pins. The Header Pin tables in Sections 7.1.1 and 7.1.2 show the full pin assignments for each header.





PocketBeagle Expansion Headers (Rev A2a)

7.1.1 P1 Header

Universal Serial Bus

SYS Power and other system control signals GPIO General purpose inputs and outputs

PRU Programmable real-time unit input, output, or peripheral CAN Controller Area Network – requires external PHY

Serial Peripheral Interface Inter-Integrated Circuit bus

UART Serial port PWM Pulse width modulator QEP Quadrature encoder peripheral

Battery

ISB

AIN Analog inputs, note that these are all enabled by default after kernel boot

Head er.Pin	Silkscr een	PocketB eagle wiring	Pr oc B all	Si P B all	Mode0 (Name)	Mode1	Mode2	Mode3	Mode4	Mode5	Mode6	Mod e7
P1.01	VIN	P1.01 (VIN)		P 1 0 & R 1 0 & T 1 0	VIN							
P1.02	A6/87	P1.02 (AIN6/ GPI087 )	A 8	C 9	ain6							
P1.02	A6/87	P1.02 (AIN6/ GPI087 )	R 5	F 2	lcd_hsync	gpmc_a 9	gpmc_a 2	pr1_edio_dat a_in3	pr1_edio_data_ out3	pr1_pru1_pr u_r30_9	pr1_pru1_pr u_r31_9	gpio 2_23
P1.03	USB1_E N	P1.03 (USB1- DRVVB US)	F 1 5	M 1 4	USB1_DRV VBUS	-	-	-	-	-	-	gpio 3_13
P1.04	89	P1.04 (PRU1.1 1)	R 6	E 1	lcd_ac_bia s_en	gpmc_a 11	pr1_mii 1_crs	pr1_edio_dat a_in5	pr1_edio_data_ out5	pr1_pru1_pr u_r30_11	pr1_pru1_pr u_r31_11	gpio 2_25
P1.05	USB1_V B	P1.05 (USB1- VBUS)	T 1 8	M 1 5	USB1_VBU S	-	-	-	-	-	-	-
P1.06	SPIO_C S	P1.06 (SPI0- CS)	A 1 6	A 1 4	spi0_cs0	mmc2_s dwp	I2C1_S CL	ehrpwm0_sy nci	pr1_uart0_txd	pr1_edio_dat a_in1	pr1_edio_dat a_out1	gpio 0_5
P1.07	USB1_V I	P1.07 (VIN- USB)		P 9 & R 9 & T 9	VIN-USB							
P1.08	SPI0_C LK	P1.08 (SPI0- CLK)	A 1 7	A 1 3	spi0_sclk	uart2_rx d	I2C2_S DA	ehrpwm0A	pr1_uart0_cts_ n	pr1_edio_sof	EMU2	gpio 0_02
P1.09	USB1 -	P1.09 (USB1- DN)	R 1 8	L 1 6	USB1_DM	-	-		-	-	-	-
P1.10	SPI0_M ISO	P1.10 (SPI0- MISO)	B 1 7	B 1 3	spi0_d0	uart2_tx d	I2C2_S CL	ehrpwm0B	pr1_uart0_rts_ n	pr1_edio_lat ch_in	EMU3	gpio 0_3
P1.11	USB1 +	P1.11 (USB1- DP)	R 1 7	L 1 5	USB1_DP	-	-		-	-	-	-
P1.12	SPI0_M	P1.12 (SPI0-	B 1	B 1	spi0_d1	mmc1_s	I2C1_S	ehrpwm0_tri	pr1_uart0_rxd	pr1_edio_dat	pr1_edio_dat	gpio

	OSI	MOSI)	6	4		dwp	DA	pzone_input		a_in0	a_out0	0_04
P1.13	USB1_I D	P1.13 (USB1- ID)	Р 1 7	L 1 4	USB1_ID	-	-	-	-	-	-	-
P1.14	+3.3V	P1.14 (VOUT- 3.3V)		F 6 & F 7 & G 6 & G 6 7	VOUT- 3.3V							
P1.15	USB1_G ND	P1.15 (GND)			GND							
P1.16	GND	P1.16 (GND)			GND							
P1.17	AIN(1.8 V)-	P1.17 (VREFN )	A 9	В 9	VREFN							
P1.1 8	AIN(1. 8V)A+	P1.18 (VREF P)	B 9	В 7	VREFP							
P1.1 9	AIN(1. 8V)0	P1.19 (AIN0- 1.8V)	B 6	A 8	ain0							
P1.2 0	20	P1.20 (PRU0. 16)	D 1 4	В 4	xdma_ev ent_intr1	-	tclkin	clkout2	timer7	pr1_pru0_p ru_r31_16	EMU3	gpio 0_20
P1.2 1	AIN(1. 8V)1	P1.21 (AIN1- 1.8V)	C 7	B 8	ain1							
P1.2 2	GND	P1.22 (GND)			GND							
P1.2 3	AIN(1. 8V)2	P1.23 (AIN2- 1.8V)	B 7	В 6	ain2							
P1.2 4	VOUT	P1.24 (VOUT -5V)		K 6 & K 7 & L 6 & L 7	VOUT-5V							
P1.2	AIN(1.	P1.25 (AIN3-	А	С	ain3							

5	8V)3	1.8V)	7	6								
P1.2 6	I2C2_S DA	P1.26 (I2C2- SDA)	D 1 8	B 1 0	uart1_cts n	timer6	dcan0_ tx	I2C2_SDA	spi1_cs0	pr1_uart0_ cts_n	pr1_edc_lat ch0_in	gpio 0_12
P1.2 7	AIN(1. 8V)4	P1.27 (AIN4- 1.8V)	C 8	C 7	ain4							
P1.2 8	I2C2_S CL	P1.28 (I2C2- SCL)	D 1 7	A 1 0	uart1_rts n	timer5	dcan0_ rx	I2C2_SCL	spi1_cs1	pr1_uart0_ rts_n	pr1_edc_lat ch1_in	gpio 0_13
P1.2 9	PRU0_ 7	P1.29 (PRU0. 7)	A 1 4	C 4	mcasp0_ ahclkx	eQEP0 _strobe	mcasp 0_axr3	mcasp1_ax r1	EMU4	pr1_pru0_p ru_r30_7	pr1_pru0_p ru_r31_7	gpio 3_21
P1.3 0	U0_TX	P1.30 (UART 0-TX)	E 1 6	B 1 2	uart0_tx d	spi1_cs 1	dcan0_ rx	I2C2_SCL	eCAP1_in_P WM1_out	pr1_pru1_p ru_r30_15	pr1_pru1_p ru_r31_15	gpio 1_11
P1.3 1	PRUO_ 4	P1.31 (PRU0. 4)	B 1 2	A 3	mcasp0_ aclkr	eQEP0 A_in	mcasp 0_axr2	mcasp1_acl kx	mmc0_sdwp	pr1_pru0_p ru_r30_4	pr1_pru0_p ru_r31_4	gpio 3_18
P1.3 2	U0_RX	P1.32 (UART 0-RX)	E 1 5	A 1 2	uart0_rx d	spi1_cs 0	dcan0_ tx	I2C2_SDA	eCAP2_in_P WM2_out	pr1_pru1_p ru_r30_14	pr1_pru1_p ru_r31_14	gpio 1_10
P1.3 3	PRUO_ 1	P1.33 (PRU0. 1)	B 1 3	A 2	mcasp0_f sx	ehrpw m0B	-	spi1_d0	mmc1_sdcd	pr1_pru0_p ru_r30_1	pr1_pru0_p ru_r31_1	gpio 3_15
P1.3 4	26	P1.34 (GPIO0 .26)	T 1 1	R 5	gpmc_ad 10	lcd_dat a21	mmc1 _dat2	mmc2_dat6	ehrpwm2_tri pzone_input	pr1_mii0_t xen	-	gpio 0_26
P1.3 5	P1.10	P1.35 (PRU1. 10)	V 5	F 1	lcd_pclk	gpmc_ a10	pru_m ii0_crs	pr1_edio_d ata_in4	pr1_edio_dat a_out4	pr1_pru1_p ru_r30_10	pr1_pru1_p ru_r31_10	gpio 2_24
P1.3 6	PWM0 A	P1.36 (PWM 0A)	A 1 3	A 1	mcasp0_ aclkx	ehrpw m0A	-	spi1_sclk	mmc0_sdcd	pr1_pru0_p ru_r30_0	pr1_pru0_p ru_r31_0	gpio 3_14

Figure 43 shows the schematic diagram for the P1 Header.

Figure 43 P1 Header

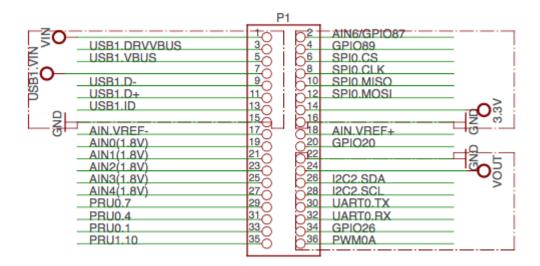


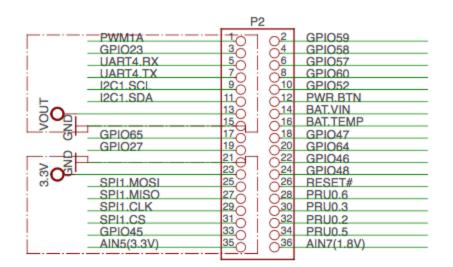
Table 7. P1 Header Pinout

Use scroll bar at bottom of chart to see additional features in columns to the right. When printing this document you will need to print this chart separately.

### 7.1.2 P2 Header

Figure 44 shows the schematic diagram for the P2 Header.

Figure 44. P2 Header



#### Table 8. P2 Header Pinout

Use scroll bar at bottom of chart to see additional features in columns to the right. When printing this document you will need to print this chart separately.

wiring Ball Ball Inforce (Name) wirder	ler.Pin	Silkscreen	PocketBeagle wiring	Proc Ball	SiP Ball	Mode0 (Name)	Mode1	Mode2
--	---------	------------	------------------------	--------------	-------------	--------------	-------	-------

1	PWM1A	P2.01 (PWM1A)	U14	P12	gpmc_a2	gmii2_txd3	rgmii2_td3
2	59	P2.02 (GPIO1.27)	V17	T16	gpmc_a11	gmii2_rxd0	rgmii2_rd0
3	23	P2.03 (GPIO0.23)	T10	Р5	gpmc_d9	lcd_data22	mmc1_dat1
4	58	P2.04 (GPIO1.26)	T16	R15	gpmc_a10	gmii2_rxd1	rgmii2_rd1
5	U1_RX	P2.05 (UART4- RX)	T17	P15	gpmc_wait0	gmii2_crs	gpmc_csn4
6	57	P2.06 (GPIO1.25)	U16	T15	gpmc_a9	gmii2_rxd2	rgmii2_rd2
7	U1_TX	P2.07 (UART4- TX)	U17	R16	gpm_ wp	gmii2_rxerr	gpmc_csn5
8	60	P2.08 (GPIO1.28)	U18	N14	gpmc_be1n	gmii2_col	gpmc_csn6
9	I2C1_SCL	P2.09 (I2C1- SCL)	D15	B11	uart1_txd	mmc2_sdwp	dcan1_rx
0	52	P2.10 (GPIO1.20)	R14	R13	gpmc_a4	gmii2_txd1	rgmii2_td1
1	I2C1_SDA	P2.11 (I2C1- SDA)	D16	A11	uart1_rxd	mmc1_sdwp	dcan1_tx
2	РВ	P2.12 (POWER_BTN)		T11	POWER		
3	VOUT	P2.13 (VOUT- 5V)		K6, K7, L6, L7	VOUT-5V		
4	BAT +	P2.14 (VIN- BAT)		P8, R8, T8	VIN-BAT		

5	GND	P2.15 (GND)			GND		
б	BAT -	P2.16 (BAT- TEMP)		N6	BAT-TEMP		
7	65	P2.17 (GPIO2.1)	V12	T7	gpmc_clk	lcd_memory_clk	gpmc_wait1
8	47	P2.18 (PRU0.15i)	U13	P7	gpmc_ad15	lcd_data16	mmc1_dat7
	27	P2.19 (GPIO0.27)	U12	Т5	gpmc_ad11	lcd_data20	mmc1_dat3
0	64	P2.20 (GPIO2.0)	T13	R7	gpmc_csn3	gpmc_a3	rmii2_crs_d
1	GND	P2.21 (GND)			GND		
2	46	P2.22 (GPIO1.14)	V13	T6	gpmc_ad14	lcd_data17	mmc1_dat6
3	+3.3V	P2.23 (VOUT- 3.3V)		F6 & F7 & G6 & G7	VOUT-3.3V		
4	48	P2.24 (GPIO1.12)	T12	P6	gpmc_ad12	lcd_data19	mmc1_dat4
5	SPI1_MOSI	P2.25 (SPI1- MOSI)	E17	C13	uart0_rtsn	uart4_txd	dcan1_rx
6	RST	P2.26 (NRESET)	A10	R11	nRESETIN_OUT	-	-
7	SPI1_MISO	P2.27 (SPI1- MISO)	E18	C12	uart0_ctsn	uart4_rxd	dcan1_tx
8	PRU0_6	P2.28 (PRU0.6)	D13	C3	mcasp0_axr1	eQEP0_index	-
9	SPI1_CLK	P2.29 (SPI1-	C18	C5	eCAP0_in_PWM0_out	uart3_txd	spi1_cs1

	CLK)					
PRU0_3	P2.30 (PRU0.3)	C12	B1	mcasp0_ahclkr	ehrpwm0_synci	mcasp0_axr
SPI1_CS	P2.31 (SPI1-CS1)	A15	A4	xdma_event_intr0	-	timer4
PRU0_2	P2.32 (PRU0.2)	D12	B2	mcasp0_axr0	ehrpwm0_tripzone_input	-
45	P2.33 (GPIO1.13)	R12	R6	gpmc_ad13	lcd_data18	mmc1_dat5
PRU0_5	P2.34 (PRU0.5)	C13	В3	mcasp0_fsr	eQEP0B_in	mcasp0_axr
A5/86	P2.35 (AIN5/GPIO86)	B8	C8	ain5		
A5/86	P2.35 (AIN5/GPIO86)	U5	F3	lcd_vsync	gpmc_a8	gpmc_a1

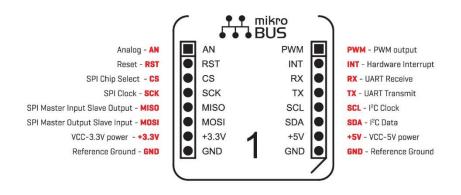
## 7.2 mikroBUS socket connections

mikroBUS and, by extension "mikroBUS Click boards", are trademarks of MikroElektronika. We do not make any claims of compatibility nor adherence to their specification. We've just seen that many of the Click boards "just work".

The Expansion Headers on PocketBeagle have been designed to accept up to two Click Boards added to the header pins at the same time. This provides an exciting opportunity to add functionality easily to PocketBeagle from <u>'hundreds of existing add-on Click Boards'</u>.

The mikroBUS standard comprises a pair of 1×8 female headers with a standardized pin configuration. The pinout (always laid out in the same order) consists of three groups of communications pins (SPI, UART and I2C), six additional pins (PWM, Interrupt, Analog input, Reset and Chip select), and two power groups (+3.3V and 5V).

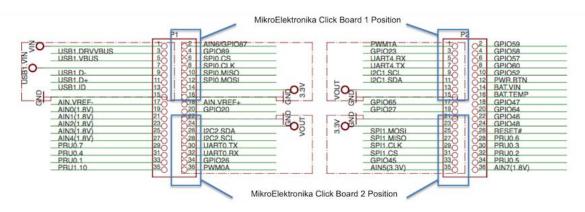
Figure 45. mikroBUS



The Expansion Header pin alignment enables 2 Click Boards on the top side of PocketBeagle using the inside rails of the headers. This leaves the outside rails open to be accessed from either the top or the bottom of

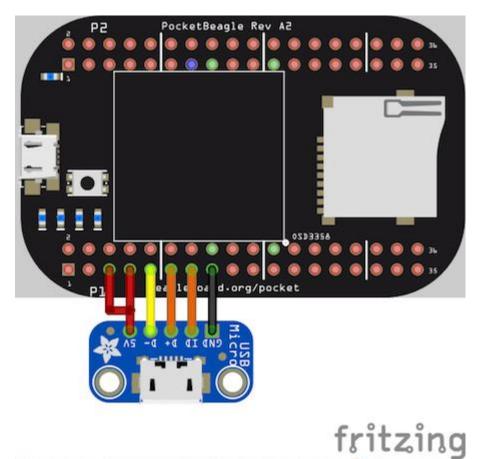
PocketBeagle. Place each Click Board into the position shown in Figure 46, with one Click Board facing each direction. When choosing Click boards, make sure you are checking that they meet the 3.3V requirements for PocketBeagle. A growing number of community members are trying out various Click Boards and posting results on the <u>'PocketBeagle Wiki mikroBus Click Boards page'</u>.

#### Figure 46. PocketBeagle Both Headers



## 7.3 Setting up an additional USB Connection

You can add an additional USB connection to PocketBeagle easily by connecting a microUSB breakout. By default in the current software, the system should be configured to use this port as a host. Keep up to date on this project on the <u>'PocketBeagle Wiki FAQ'</u>.



## 8.0 PocketCape Support

This is a placeholder for recommendations for those building their own PocketCape designs. If you'd like to join the conversation <u>'check out the discussion on the google group for PocketBeagle'</u>

## 9.0 PocketBeagle Mechanical

## 9.1 Dimensions and Weight

Size: 2.21" x 1.38" (56mm x 35mm)

Max height: .197" (5mm)

PCB size: 55mm x 35mm

PCB Layers: 4

PCB thickness: 1.6mm

**RoHS Compliant: Yes** 

Weight: 10g

Rough model can be found at *github.com/beagleboard/pocketbeagle/tree/master/models* 

## **10.0 Additional Pictures**

Figure 47. PocketBeagle Front BW

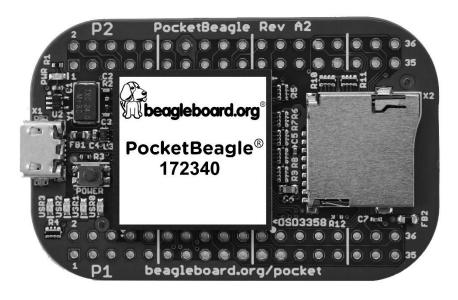
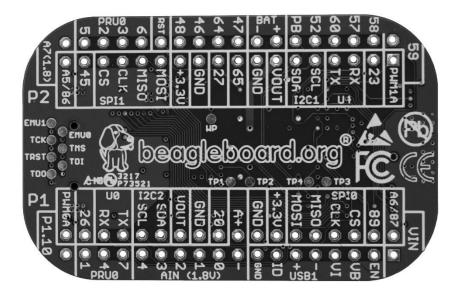


Figure 48. PocketBeagle Back BW



## **11.0 Support Information**

All support for this design is through the BeagleBoard.org community at:

- <u>beagleboard@googlegroups.com</u> or
- <u>beagleboard.org/discuss</u>.

### 11.1 Hardware Design

Design documentation can be found on the wiki. <u>https://github.com/beagleboard/pocketbeagle</u>Including:

- Schematic in PDF<u>https://github.com/beagleboard/pocketbeagle/blob/master/PocketBeagle\_sch.pdf</u>
- Schematic and layout in EAGLE <u>https://github.com/beagleboard/pocketbeagle/tree/master/EAGLE</u>
- Schematic and layout in KiCAD <u>https://github.com/beagleboard/pocketbeagle/tree/master/KiCAD</u>
- Bill of Materials <u>https://github.com/beagleboard/pocketbeagle/blob/master/PocketBeagle\_BOM.csv</u>
- System Reference Manual https://github.com/beagleboard/pocketbeagle.

### **11.2 Software Updates**

It is a good idea to always use the latest software. Instructions for how to update your software to the latest version can be found at:

Download the latest software files from *beagleboard.org/latest-images* 

## **11.5 Export Information**

- ECCN: EAR99
- CCATS: G173833
- Documentation: <u>github.com/beagleboard/pocketbeagle/blob/master/regulatory/PocketBeagle\_Export\_Cl</u>
   <u>assification.pdf</u>

## **11.4 RMA Support**

If you feel your board is defective or has issues and before returning merchandise, please seek approval from the manufacturer using *beagleboard.org/support/rma*. You will need the manufacturer, model, revision and serial number of the board.

## **11.5 Getting Help**

If you need some up to date troubleshooting techniques, the Wiki is a great place to start *<u>github.com/beagleboard/pocketbeagle/wiki</u>*.

If you need professional support, check out *beagleboard.org/resources*.