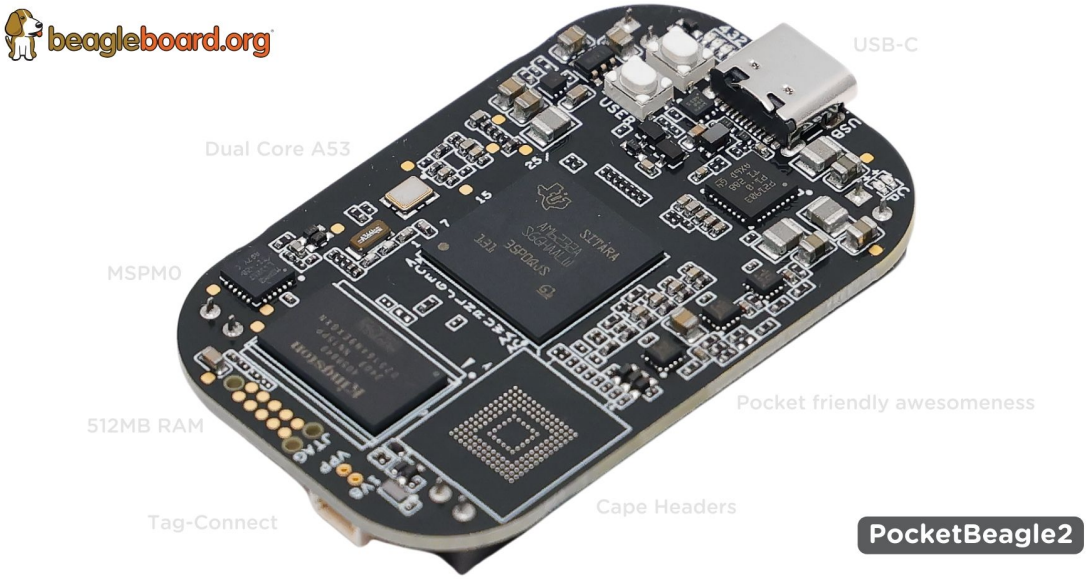
PocketBeagle 2

BeagleBoard.org Foundation
Jan 22, 2025

Table of contents

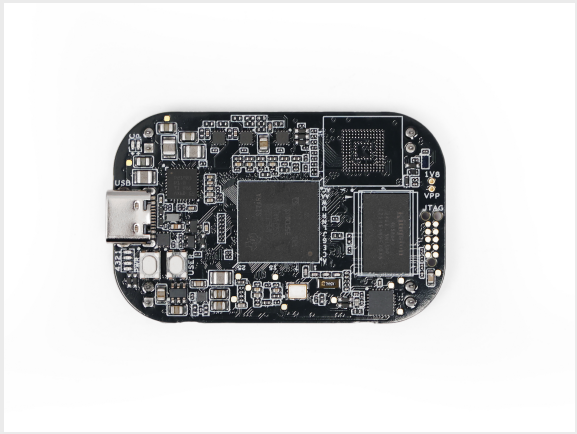
PocketBeagle 2 is an upgraded version of the widely popular PocketBeagle, designed as an ultra-compact, low-cost, and powerful single-board computer (SBC). Targeted at developers, students, and hobbyists, PocketBeagle 2 retains the simplicity and flexibility of its predecessor while delivering enhanced performance and expanded features to support modern development needs. PocketBeagle 2 is ideal for creating IoT devices, robotics projects, and educational applications. Its small form factor and low power consumption make it a versatile platform for embedded development, whether prototyping or deploying at scale.

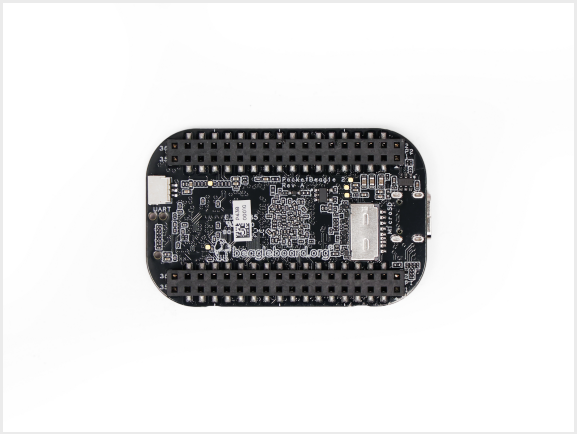


Chapter 1

Introduction

PocketBeagle 2 is based on Texas Instrments AM6232 SoC, it's dual A53 cores can provides higher performance than classic PocketBeagle. The new design comes with pre-soldered headers, 3-pin JST-SH 1.00mm uart debug port, USB-C port, MSPM0L1105, 512MB RAM, and LiPo Battery charger.





1.1 Comparison

The board is intended to provide functionality well beyond classic PocketBeagle, while still providing compatibility with PocketBeagle's expansion headers as much as possible. There are several significant differences between the designs.

Table 1.1: Table: PocketBeagle comparison

Feature	PocketBeagle 2	PocketBeagle original
SoC	AM6232	AM3358
Arm CPU	Cortex-A53 (64-bit)	Cortex-A8 (32-bit)
Arm cores	2 x 1GHz	1 x 1GHz
RAM	512MB DDR4	512MB DDR3

1.2 PocketBeagle 2 Features and Specification

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

Table 1.2: Table: PocketBeagle 2 Features and Specification

	Feature
Processor	Texas Instruments AM6232
SDRAM Memory	LPDDR4 3200MHz (512MB) Kingston D2516AN9EXGXN-TU
PMIC	TPS6521903
Debug Support	3 pin 3.3V JST-SH 1.00mm UART debug port (RPI debug probe compatible)
	10-pin JTAG TAG-CONNECT footprint
Power Source	USB C or Cape Header VIN (5V @ 1A)
PCB	55 x 35 mm
Indicators	1x Power, 1x Battery charging, and 4x User Controllable LEDs
SD/MMC Connector	microSD (1.8/3.3V)
User Input	1. Power Button 2. User/Boot Button
Weight	12.7gm

1.3 Board Component Locations

This section describes the key components on the board. It provides information on their location and function. Familiarize yourself with the various components on the board.

1.3.1 Board components

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

Front components location

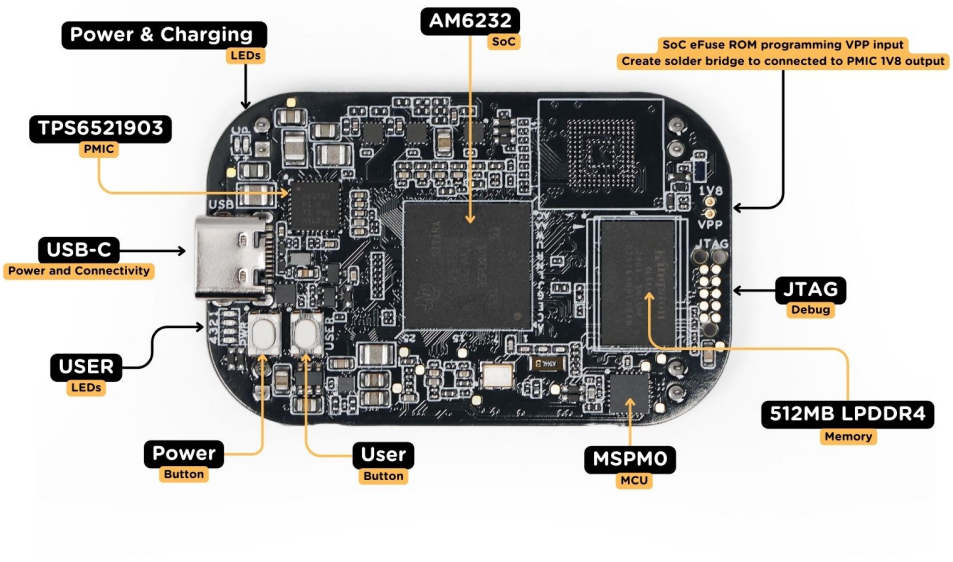


Fig. 1.1: PocketBeagle 2 board front components location

Table 1.3: PocketBeagle 2 board front components location table

Feature	Description
AM6232 SoC	Internet of Things (IoT) and gateway SoC with dual core A53 @ 1GHz
MSPM0 MCU	MSPM0 MCU to provide ADC and EEPROM functionality
U, P and C LEDs	USR1 - USR4 (U) user LEDs, Power (P) & Charging (C) LED indicator
USB C	Power and connectivity.
User button	User action button, hold down to boot from sdCard on a board with eMMC
Power button	Hold down to toggle ON/OFF
TPS6521903	Power Management Integrated Circuit (PMIC)
512MB RAM	512MB DDR4 RAM
JTAG debug port	Tag-Connect JTAG (AM6232) debug port

Back components location

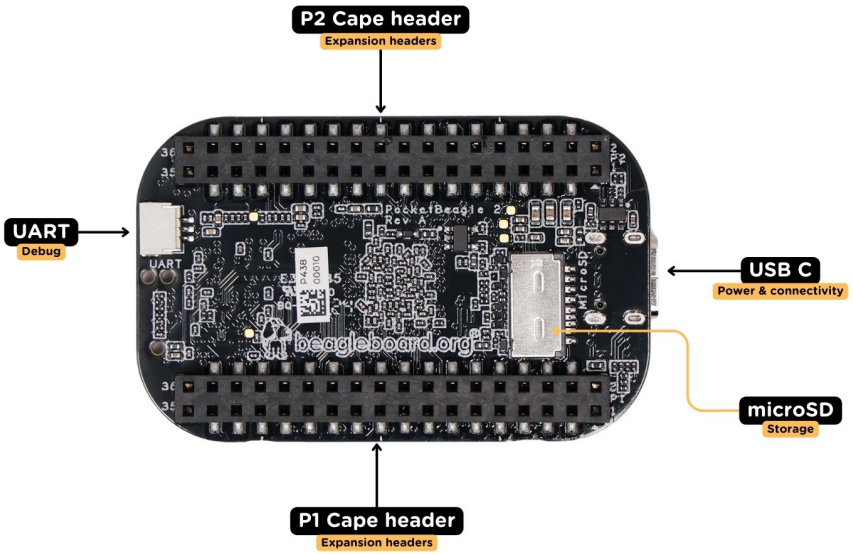


Fig. 1.2: PocketBeagle 2 board back components location

Table 1.4: PocketBeagle 2 board back components location table

Feature	Description
microSD	Micro SD Card holder
P1 & P2 cape header	Expansion headers for PocketBeagle capes.
UART debug ports	3pin JST-SH 1.00mm UART debug port (RPI debug probe compatible)

Chapter 2

Quick Start Guide

This section provides instructions on how to hook up your board. This Beagle requires a 5V @ 1A (5W) power supply to work properly via either USB Type-C power adapter or via cape header pins.

Recommended adapters can be found at accessories-power-supplies section.

2.1 What's In the Box

In the box you will find two main items,

- PocketBeagle 2
- Instruction card

Note: A USB-C to USB-C / USB-A to USB-C cable is not included, but recommended for the tethered scenario and creates a developer experience where the board can be used immediately with no other equipment needed.

Tip: For board files, 3D model, and more, you can checkout PocketBeagle 2 repository on OpenBeagle.

2.2 Creating bootable microSD card

To get started with creating a bootable microSD card, you need folloiwing items,

1. Up to date bb-imager-rs
2. microSD card (8GB or larger)

Tip: If required you can manually download the image from beagleboard.org distros page, for example, https://www.beagleboard.org/distros/pocketbeagle2-debian-12-9-2025-01-15-minimal

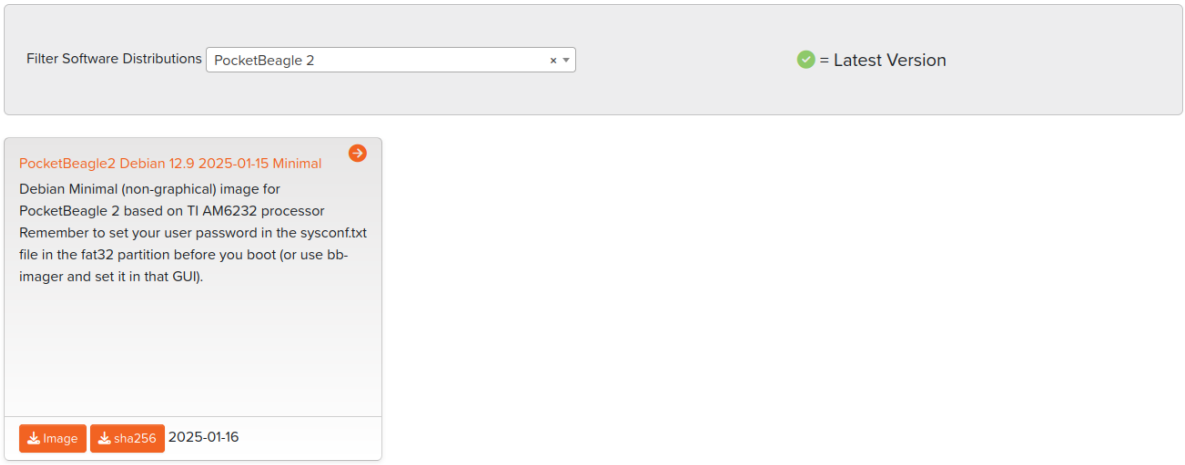


Fig. 2.1: Distros selection

To begin you have to select PocketBeagle 2 from the list of boards, then select the image you want to flash, and finally select the microSD card. After doing this you have to select the config button to configure the image and then click on the flash button to start the flashing process.

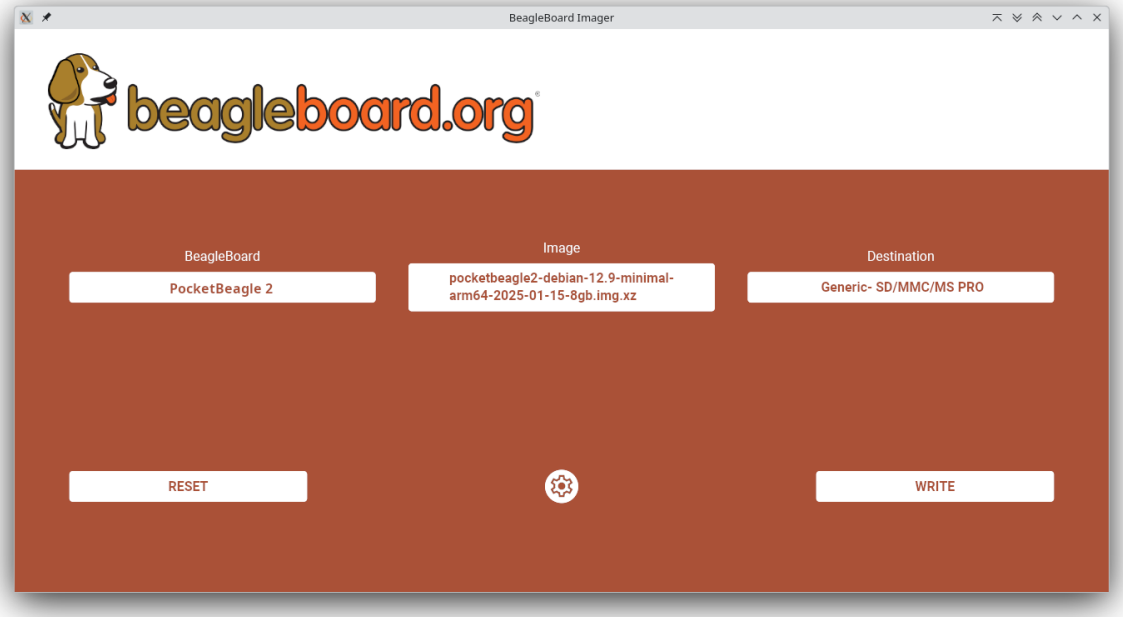


Fig. 2.2: Board and image selection

Below image shows the configuration options available for the image.

Tip: You can enable Skip `Verification` as shown in the image below, which will make the flashing process faster by skipping the verification step.

Make sure to select the correct microSD card and click on the flash button to start the flashing process.

Once the flashing is done, you will see the following screen.

Fig. 2.3: Configuration options

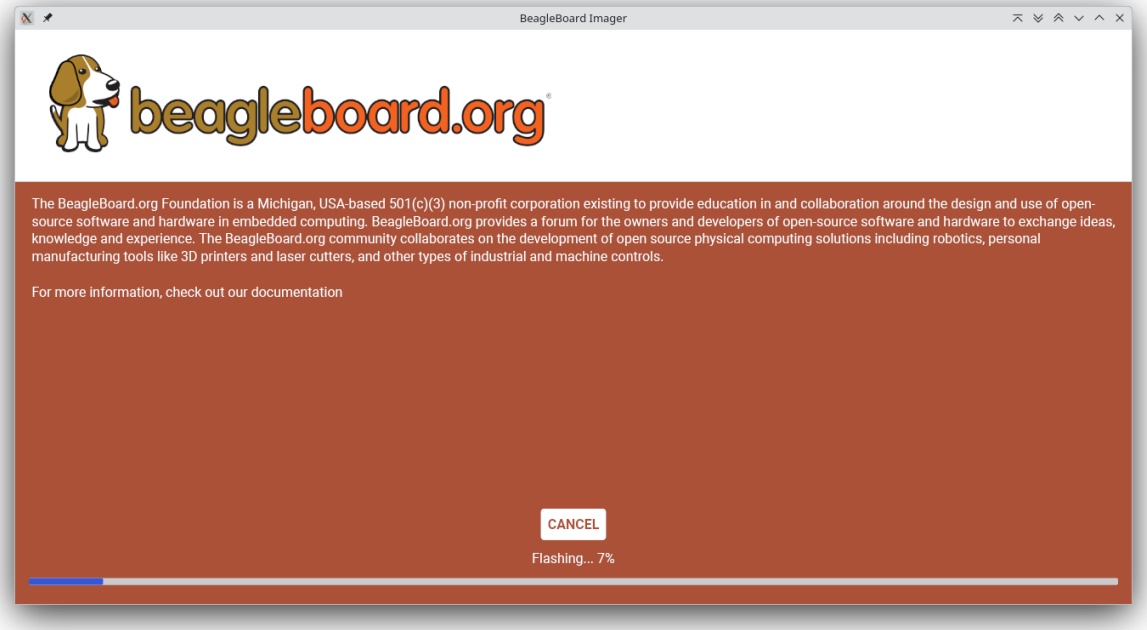


Fig. 2.4: Flashing in progress

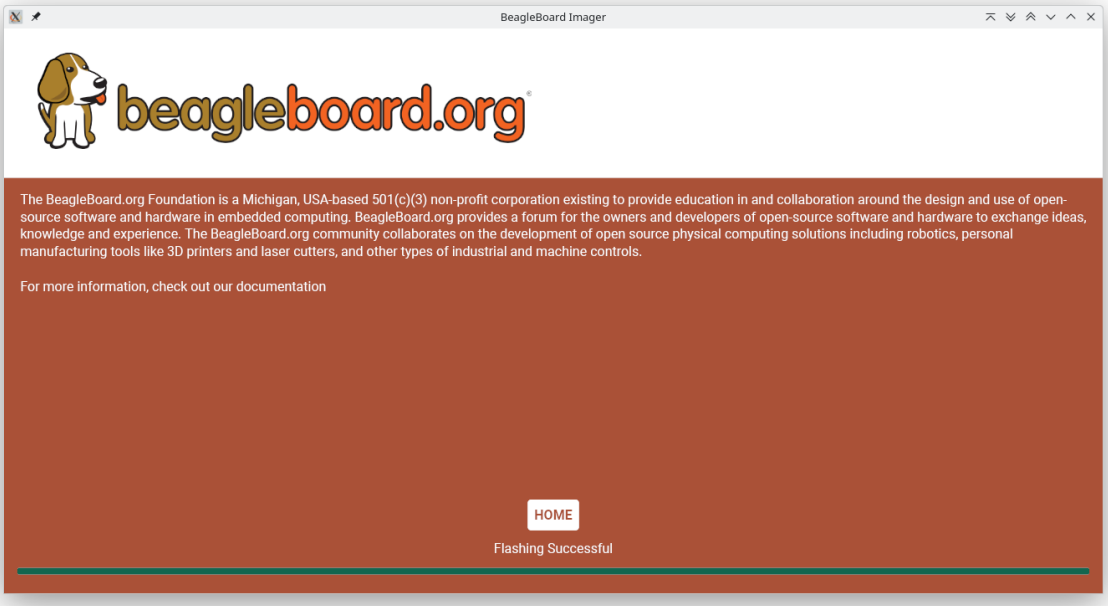


Fig. 2.5: Flashing done

Your microSD card is now ready to boot PocketBeagle 2.

2.3 Methods of operation

1. Directly tethered to a PC via USB-C port.
2. With TechLab Cape for sesors, USB host, LEDs and Buttons.

2.3.1 Main Connection Scenarios

This section describes how to connect and power the board and serves as a slightly more detailed description of the Quick Start Guide included in the box. The board can be configured in several different ways, but we will discuss the two most common scenarios.

- Tethered to a PC via the USB cable
- Attached to a cape like TechLab Cape

Chapter 3

Design and Specifications

In this chapter, we delve into the intricate design and detailed specifications of PocketBeagle 2, offering a thorough understanding of its hardware architecture. We begin with a high-level overview, presenting block diagrams that illustrate the main components and their interconnections, including the System on Chip (SoC), power management, memory, connectivity interfaces, and peripheral components. These diagrams provide a visual representation of the I2C tree, power distribution, and boot configurations, essential for grasping the board's functionality.

The chapter then focuses on the heart of PocketBeagle 2, the AM6232 SoC. We explore its internal architecture, highlighting the dual ARM Cortex-A53 cores, Cortex-M4F core, and various integrated peripherals. Detailed figures illustrate the SoC's functional blocks, decoupling capacitors, DDR controller, and power management, emphasizing the importance of each component in ensuring efficient and reliable operation.

Connectivity and expansion options are also covered extensively. We discuss the USB connections, cape headers, and the MicroSD card slot, which enhance the board's versatility and usability. Additionally, we provide insights into the debug ports, including the serial debug port and JTAG connections, which are crucial for development and troubleshooting.

The power management section details the integrated circuits responsible for stable and efficient power delivery, such as the TPS6521903 PMIC, TLV62595 step-down converter, LM73100 power path management IC, and BQ21040 battery charger. Each component's role in maintaining power integrity and optimizing consumption is explained, supported by relevant figures.

Finally, we present the mechanical specifications of PocketBeagle 2, including its dimensions, weight, and PCB details. This comprehensive overview ensures that you have a complete understanding of PocketBeagle 2's design, capabilities, and potential applications, making it an invaluable resource for developers and engineers.

3.1 Block Diagram and Overview

The figure below provides a high-level overview of PocketBeagle 2 hardware architecture, illustrating the main components and their interconnections. This includes the System on Chip (SoC), power management, memory, connectivity interfaces, and other peripheral components.

- **System on Chip (SoC)**: At the core of PocketBeagle 2 is the AM6232 SoC, which integrates dual ARM Cortex-A53 cores, a Cortex-M4F core, and various peripherals. This SoC is optimized for power efficiency and performance, making it suitable for a wide range of embedded applications.
- **Power Management: The diagram highlights several power management ICs:**
  - **TPS6521903 PMIC**: Manages multiple power rails, including buck converters and LDOs, to supply necessary voltages.
  - **TLV62595 Step-Down Converter**: Provides a stable 3.3V power supply with high efficiency.
  - **LM73100 Power Path Management IC**: Seamlessly switches between multiple power sources (VIN_5V, USB_5V, VBAT) to ensure stable system voltage.

- **Memory Components: PocketBeagle 2 includes:**
  - **512MB LPDDR4 RAM**: Ensures efficient data transfer and memory access.
- **Connectivity Interfaces: The board offers various connectivity options:**
  - **USB Ports**: For data transfer and power supply.
  - **Cape Headers**: P1 and P2 headers for expansion and additional peripherals.
  - **MicroSD Card Slot**: For additional storage and boot options.
- **Debug Ports: Essential for development and troubleshooting:**
  - **Serial Debug Port**: Compatible with the Raspberry Pi Debug Probe for UART communication.
  - **JTAG**: For in-depth debugging and programming.

This comprehensive block diagram is essential for understanding the intricate design and functionality of PocketBeagle 2, providing a visual representation of how each component interacts within the system.

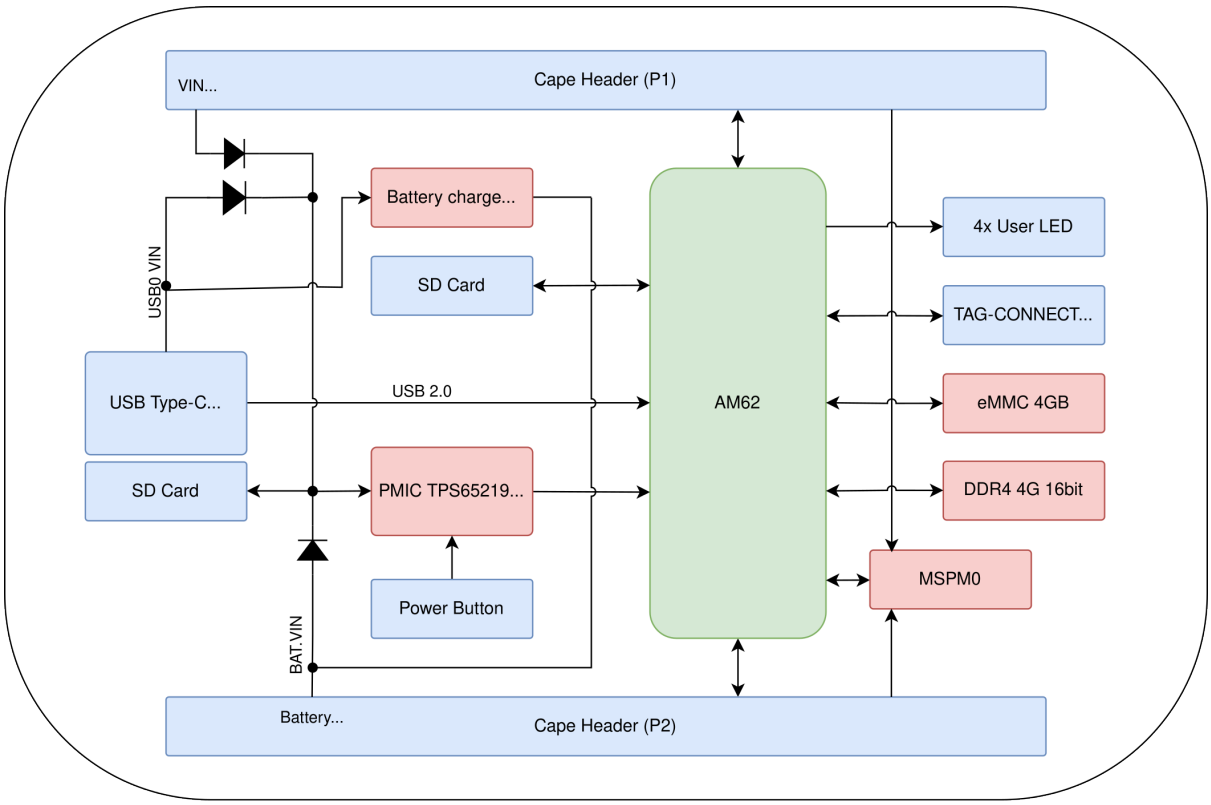


Fig. 3.1: PocketBeagle 2 Block Diagram

The following figure illustrates the I2C tree of PocketBeagle 2, showing the connections between the I2C master and various I2C slave devices on the board. The I2C tree is crucial for understanding the communication pathways and how different components interact with each other.

Key I2C Ports and Connections:

Table 3.1: Key I2C Ports and Connections

I2C	Connection
WKUP_I2C(	Connected to the TPS6521903 PMIC for power management control and monitoring.
I2C0	Connected to the MSPM0L1105 microcontroller, which emulates an 8-channel 12-bit ADC and a 4KB EEPROM.
I2C1	Available on the P1 cape header for additional peripherals and expansion options.
I2C2	Also available on the P1 cape header for additional peripherals and expansion options.
I2C3	Available on the P2 cape header for user-defined peripherals and custom applications.
MCU_I2C0	Not connected to anything internally, thus can be used if someone wants to modify PocketBeagle 2 design to add something like a QWIIC connector.

These connections ensure efficient communication and control across PocketBeagle 2, enabling robust and flexible system design.

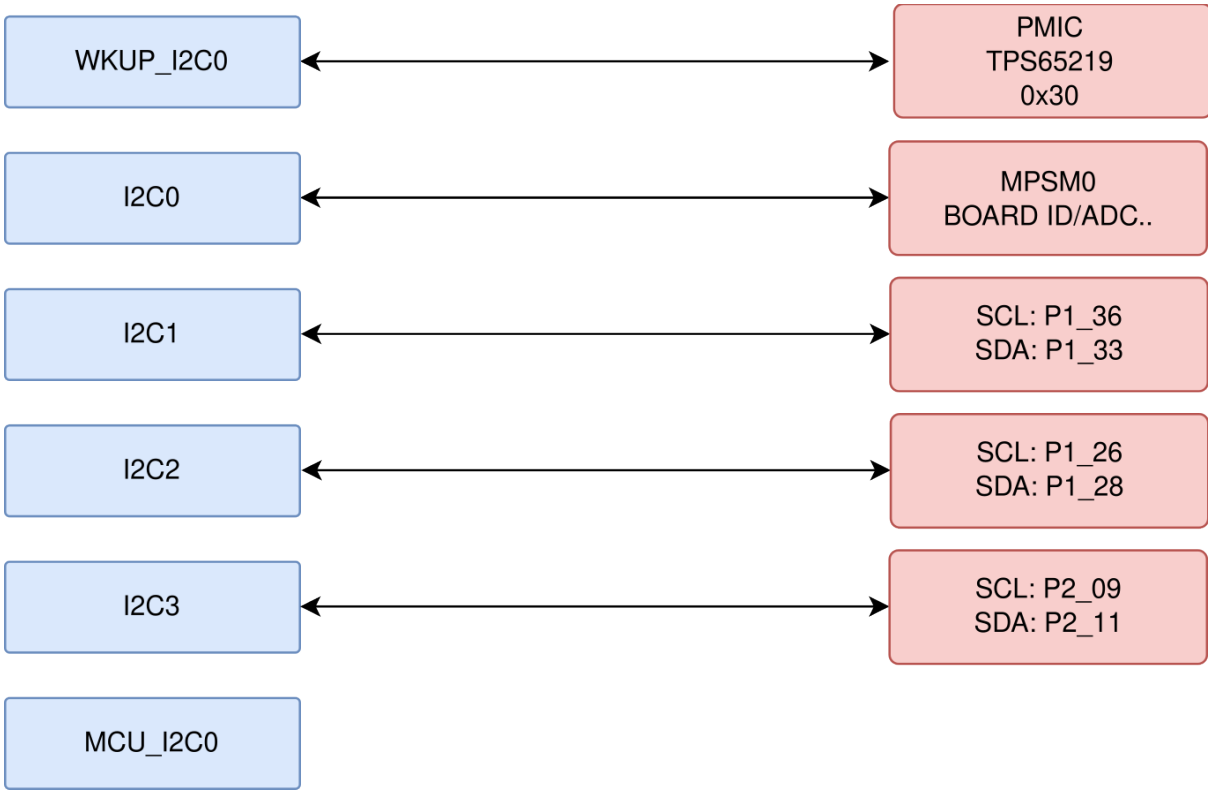


Fig. 3.2: I2C tree

The following figure shows the power tree of PocketBeagle 2, detailing the power distribution from the main power sources to various components on the board. This diagram is crucial for understanding how power is managed and distributed across the board to ensure stable and efficient operation.

Key Components and Power Paths:

- **VIN_5V**: This is the primary power input, typically supplied by an external power adapter. It is the main source of power for the board when available.
- **USB_5V**: This input comes from a USB connection. It serves as an alternative power source when VIN_5V is not available, allowing the board to be powered via a USB connection.
- **VBAT**: This is the battery voltage input, used when neither VIN_5V nor USB_5V is available. It ensures that the board remains powered in portable applications.

Power Management ICs:

- **TPS6521903 PMIC**: Manages multiple power rails, including buck converters and LDOs, to supply necessary voltages to various components. It ensures stable and efficient power delivery.

- **TLV62595 Step-Down Converter**: Provides a stable 3.3V power supply with high efficiency, powering critical components on the board.
- **LM73100 Power Path Management IC**: Seamlessly switches between VIN_5V, USB_5V, and VBAT to ensure a stable system voltage (VSYS). It prioritizes the highest available power source and transitions smoothly between sources to prevent power interruptions.
- **BQ21040 Battery Charger**: Manages the charging of a single-cell Li-Ion or Li-Polymer battery. It features high-accuracy voltage regulation, programmable charge current, and thermal protection, ensuring safe and efficient battery charging.

Power Distribution:

- **3.3V Rail**: Powers various components, including the SoC, memory, and peripheral interfaces. The TLV62595 step-down converter ensures a stable 3.3V supply.
- **1.8V and 1.2V Rails**: These lower voltage rails are generated by the TPS6521903 PMIC and are used to power specific components that require lower operating voltages.
- **VDDA 0.85V**: This rail powers the analog components of the SoC, ensuring precise analog signal processing.
- **VDD Core**: Powers the core logic of the SoC, ensuring stable operation of the processor and integrated peripherals.

The power tree diagram provides a comprehensive overview of how power is distributed and managed across PocketBeagle 2, highlighting the role of each power management component in maintaining system stability and efficiency.

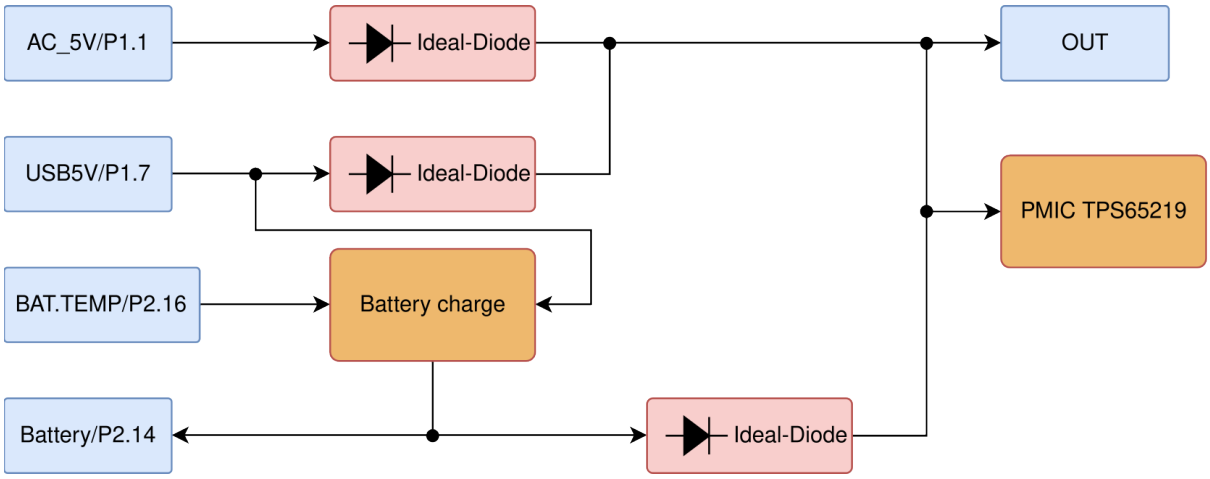


Fig. 3.3: Power tree

3.2 System on Chip (SoC)

PocketBeagle 2 is powered by the AM6232 SoC, which is a high-performance, low-power processor designed for embedded applications. The AM6232 integrates dual ARM Cortex-A53 cores, a Cortex-M4F core, and various peripherals to support a wide range of functionalities. It is optimized for power efficiency and performance, making it suitable for applications requiring robust processing capabilities while maintaining low power consumption. The AM6232 SoC functional block diagram below provides a detailed view of the internal architecture of the System on Chip. It highlights the various functional blocks such as the CPU cores, memory controllers, peripheral interfaces, and other integrated components. This diagram is essential for understanding how the SoC manages data flow and interacts with other hardware components on PocketBeagle 2 board.

Decoupling capacitors are used to filter out noise and provide a stable power supply to the SoC. They help in maintaining the integrity of the power signals by smoothing out voltage fluctuations and transient spikes,

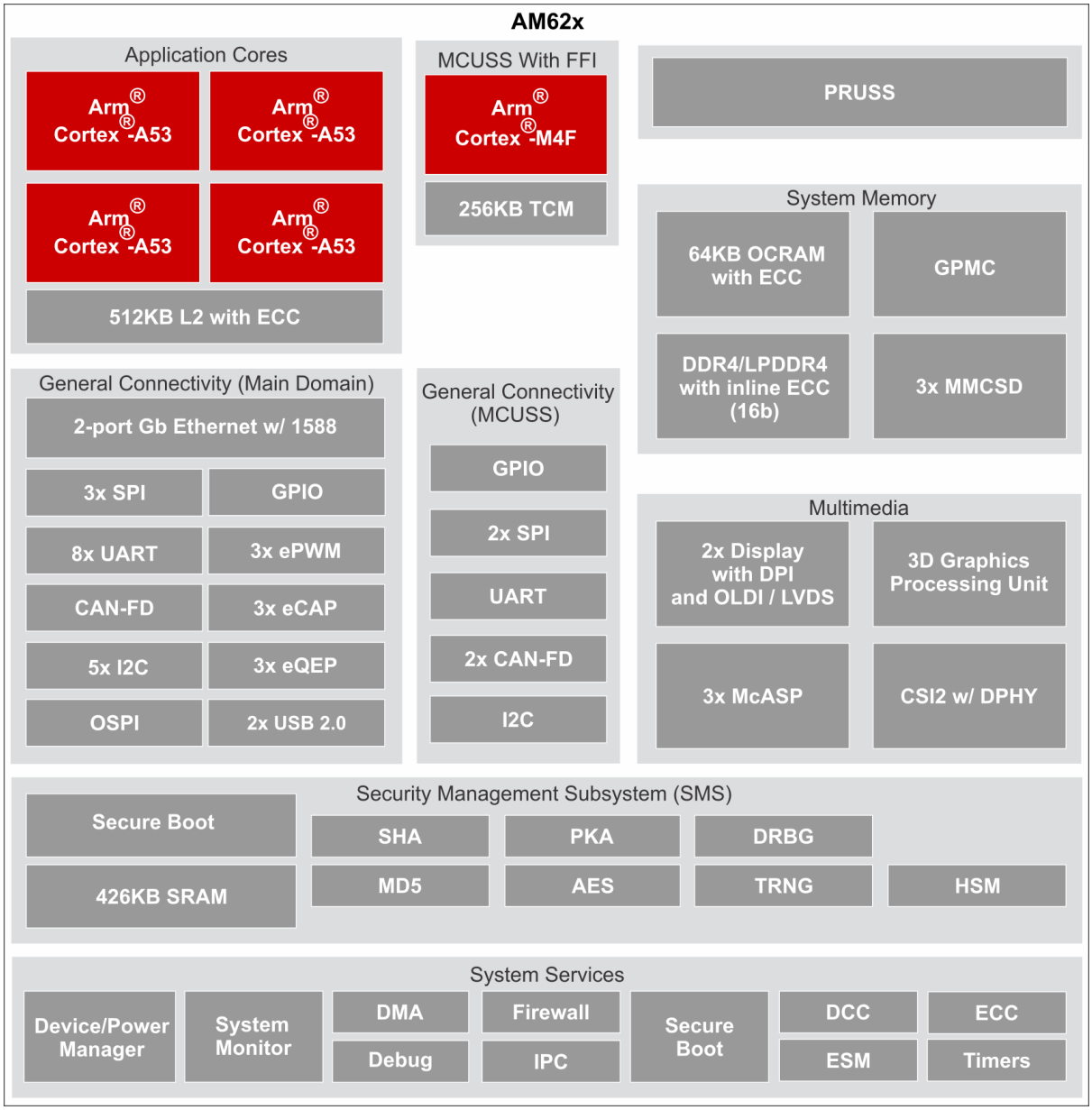


Fig. 3.4: SoC functional block diagram

ensuring reliable operation of the SoC and preventing potential malfunctions due to power instability.

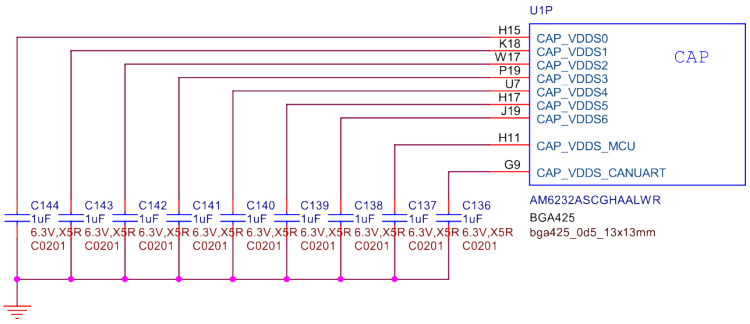


Fig. 3.5: SoC decoupling capacitors

The following figure shows the DDR controller of the SoC, which manages the communication between the processor and the DDR memory. It ensures efficient data transfer and memory access, playing a crucial role in the overall performance of the system.

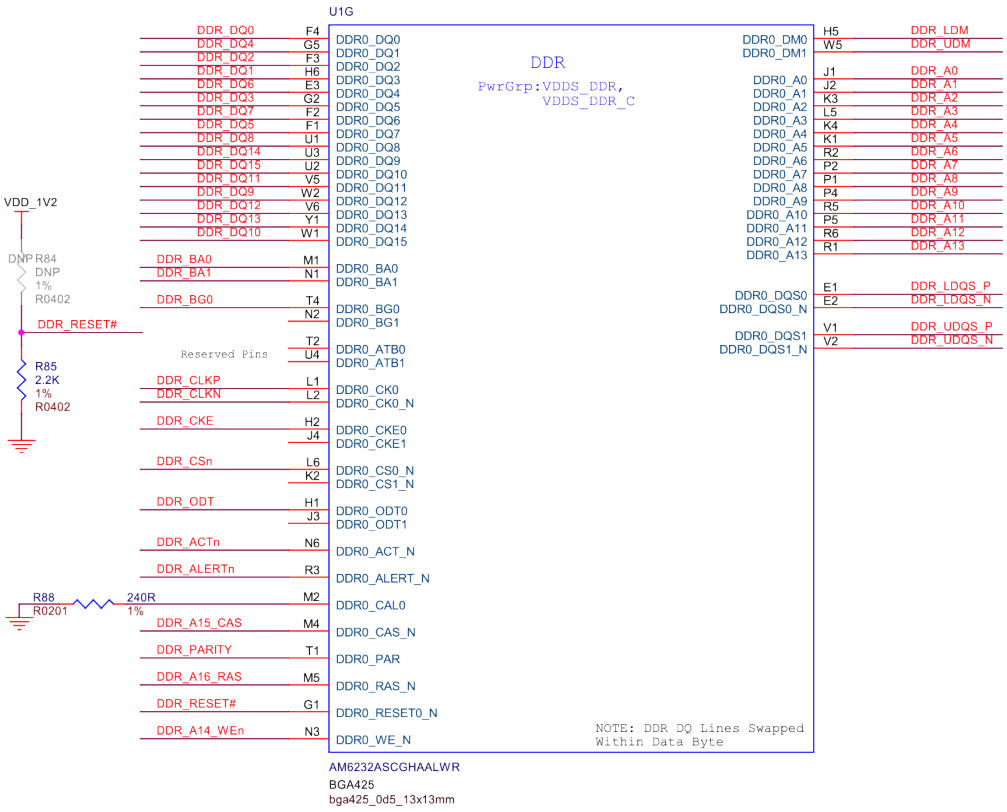


Fig. 3.6: SoC DDR controller

The following figure shows the power capacitors used for the SoC. These capacitors are crucial for maintaining stable power delivery to the SoC, filtering out noise, and ensuring reliable operation by smoothing out voltage fluctuations.

The following figure shows the power distribution for the SoC, detailing how power is supplied to various components within the SoC to ensure stable and efficient operation.

The following figure shows the VSS (Ground) connection for the SoC. This connection is crucial for providing a common reference point for all the electrical signals and ensuring the proper operation of the SoC by stabilizing the voltage levels.

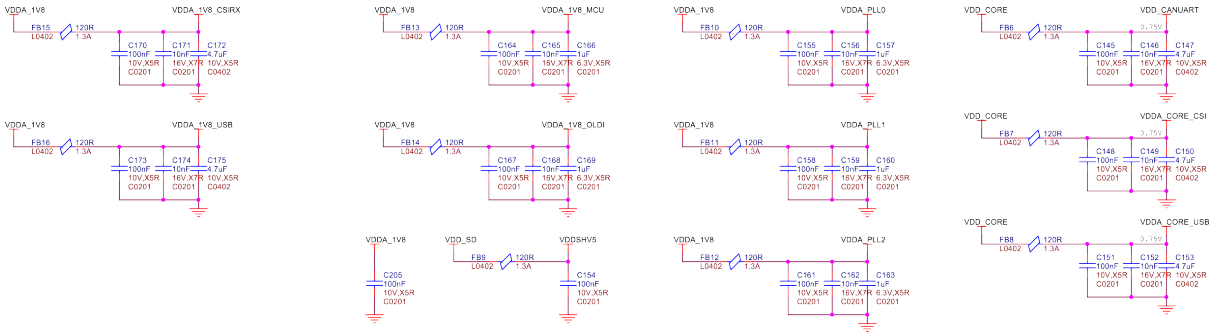


Fig. 3.7: SoC power capacitors

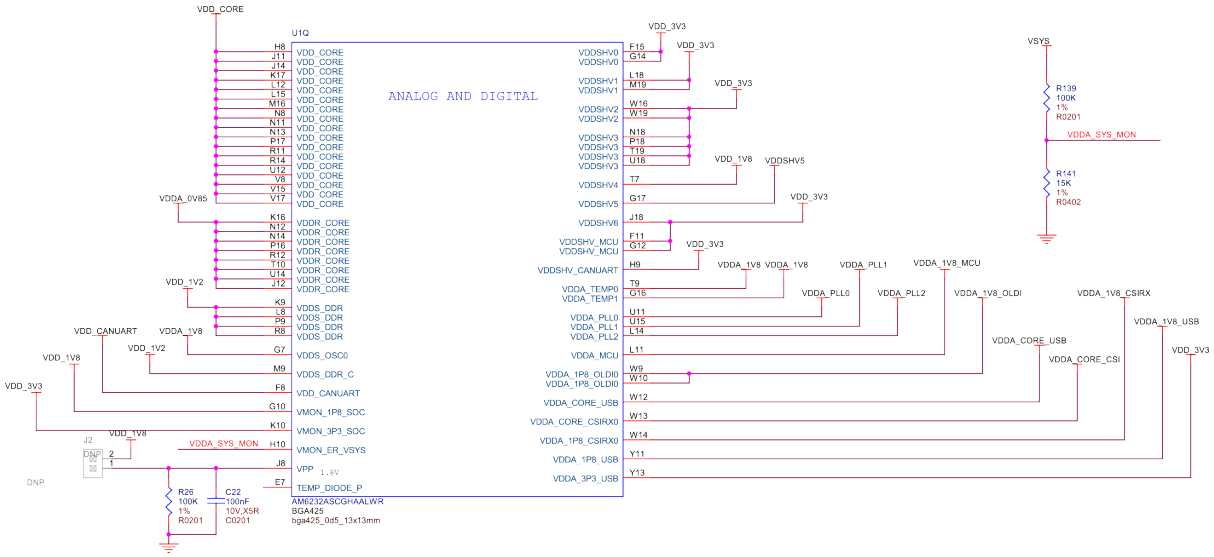


Fig. 3.8: SoC power

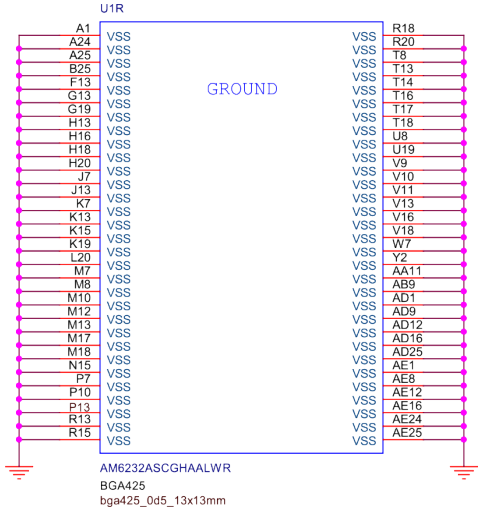


Fig. 3.9: SoC VSS (Ground) connection

3.2.1 Boot Modes

The following figure shows the boot configuration of PocketBeagle 2.

```
EMMC Version
Button Not-pressd:
1, PLL Config    B[2:0] = 0b011     : Ref Colck -> 25MHz
2, Primarry Boot B[9:3] = 0b1001001 : eMMC Boot
3, Backup Boot   B[13:10] = 0b0001  : USB DFU Boot

Button Pressd:
1, PLL Config    B[2:0] = 0b011     : Ref Colck -> 25MHz
2, Primarry Boot B[9:3] = 0b1001000 : SDCard FS Boot
3, Backup Boot   B[13:10] = 0b1011  : UART Boot


SD Version
Button Not-pressd:
1, PLL Config    B[2:0] = 0b011     : Ref Colck -> 25MHz
2, Primarry Boot B[9:3] = 0b1001000 : SDCard FS Boot
3, Backup Boot   B[13:10] = 0b0001  : USB DFU Boot

Button Pressd:
1, PLL Config    B[2:0] = 0b011     : Ref Colck -> 25MHz
2, Primarry Boot B[9:3] = 0b1001000 : SDCard FS Boot
3, Backup Boot   B[13:10] = 0b1011  : UART Boot
```

Fig. 3.10: Boot configuration

The following figure illustrates the bootstrap pins connection, which are used to select the boot mode during the power-up sequence.

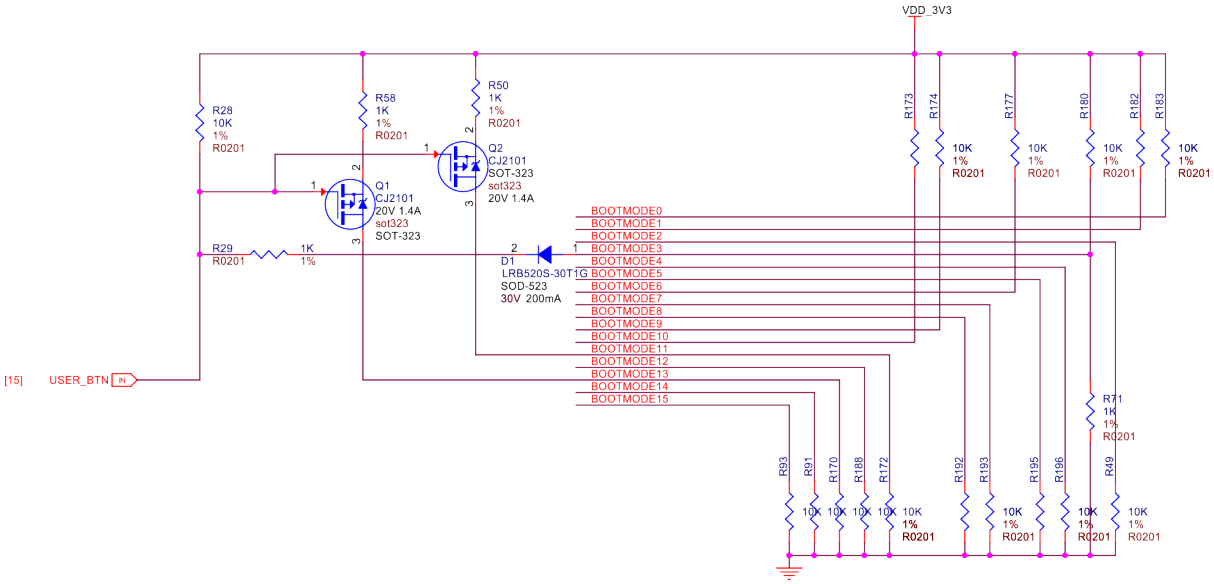


Fig. 3.11: Bootstrap pins connection

3.2.2 SoC GPIOs

GPIO GPMC

GPIO MCASP0

GPIO OSC0

GPIO OSPI

GPIO RGMII1

GPIO RGMII2

GPIO VOUT0

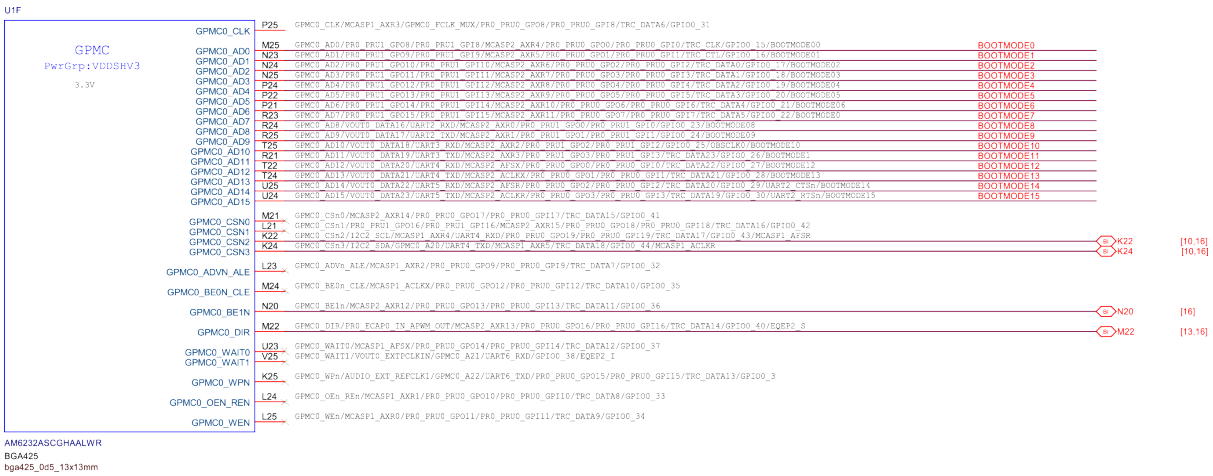


Fig. 3.12: GPIO GPMC

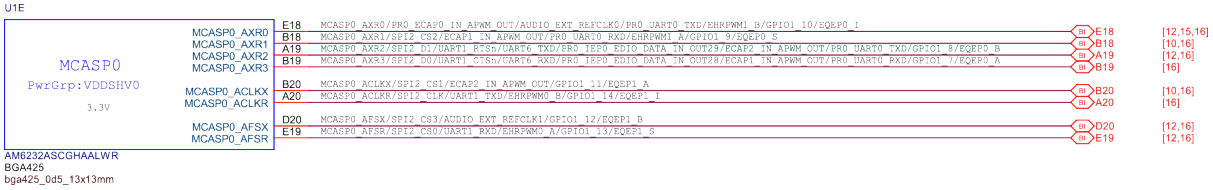


Fig. 3.13: GPIO MCASP0

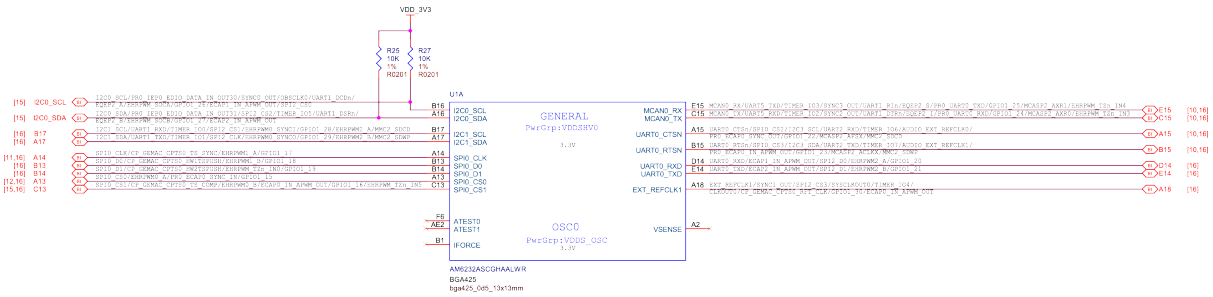


Fig. 3.14: GPIO OSC0

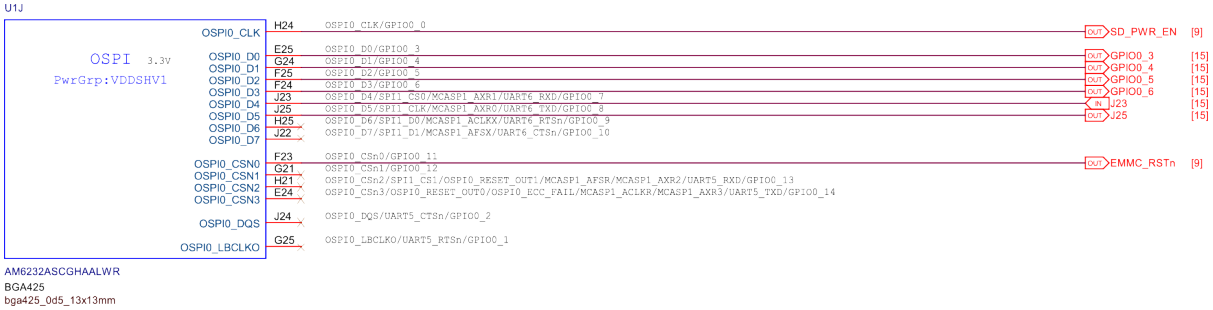


Fig. 3.15: GPIO OSPI

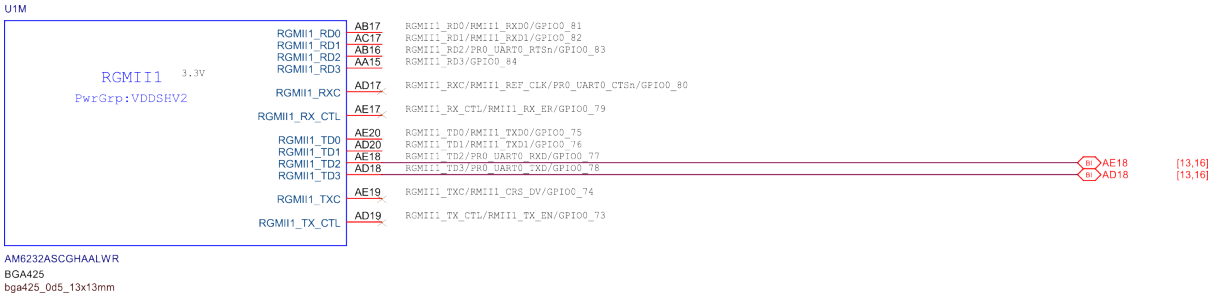


Fig. 3.16: GPIO RGMII1

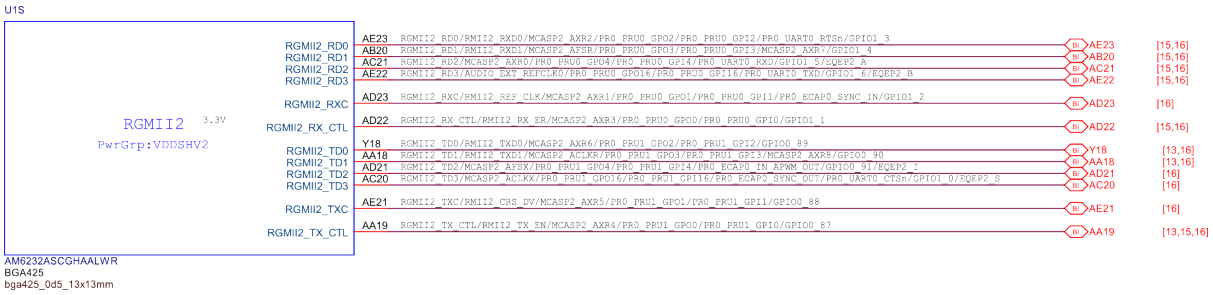


Fig. 3.17: GPIO RGMII2

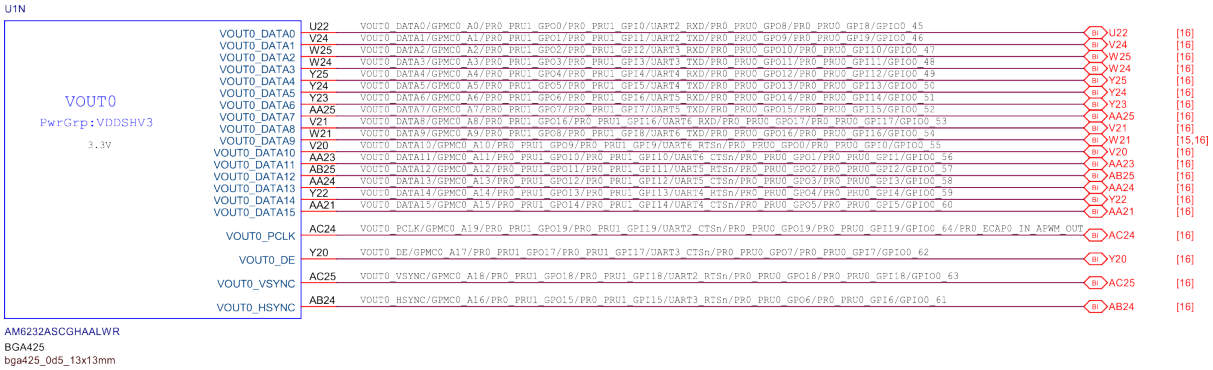


Fig. 3.18: GPIO VOUT0

MCU domain

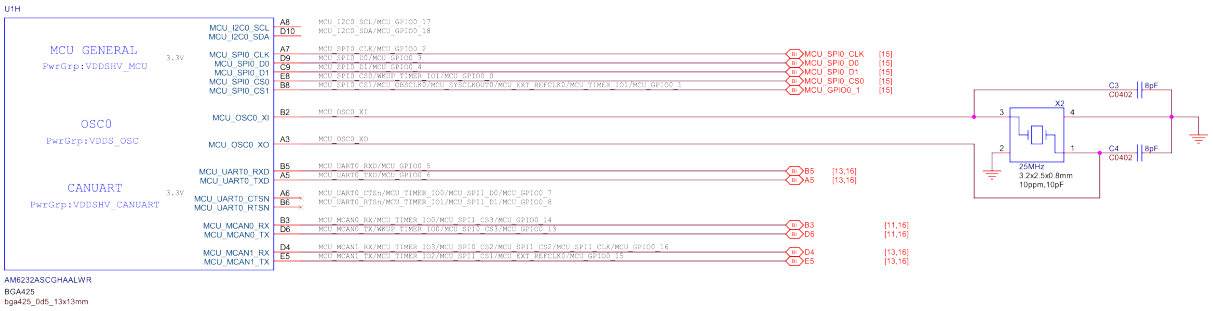


Fig. 3.19: MCU domain

MCU system

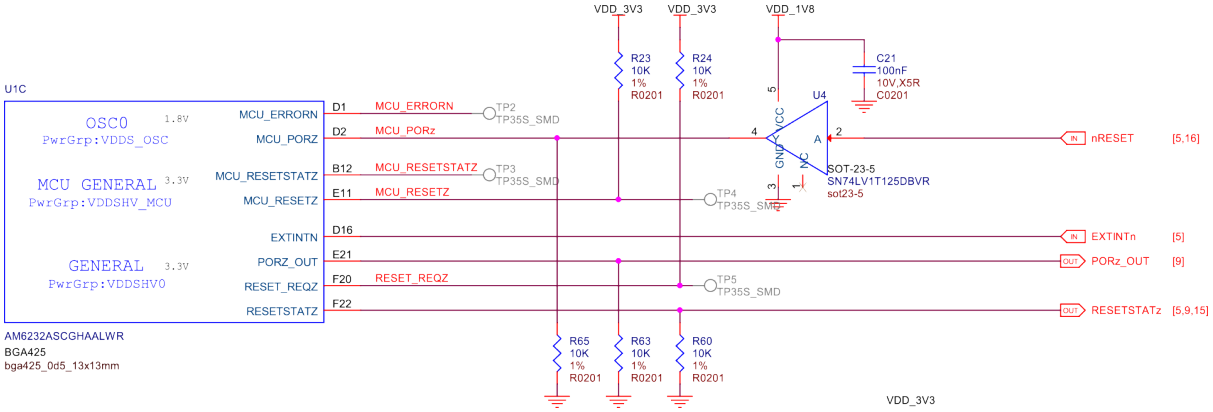


Fig. 3.20: MCU system

Wakeup domain

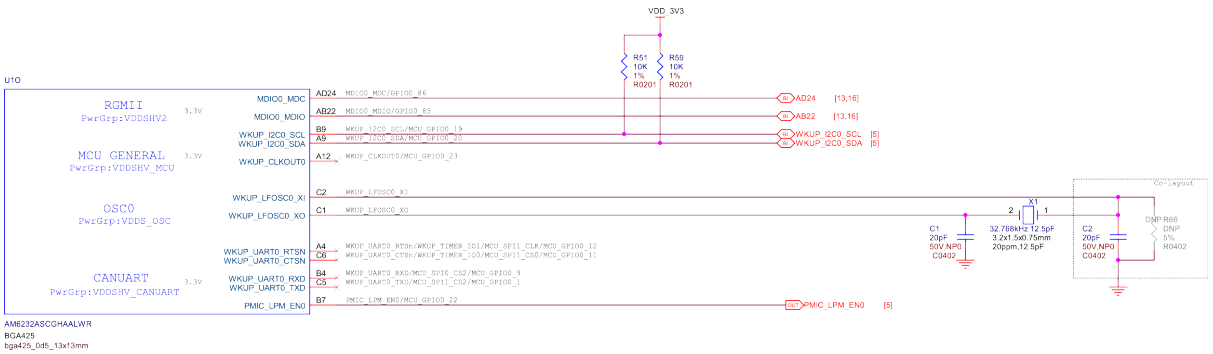


Fig. 3.21: Wakeup domain

3.3 Power Management

3.3.1 PMIC

The TPS6521903 is a power management integrated circuit (PMIC) designed to provide efficient power management for PocketBeagle. It integrates multiple power rails, including buck converters and LDOs, to supply the necessary voltages to various components on the board. The PMIC ensures stable and reliable power delivery, optimizing power consumption and extending battery life.

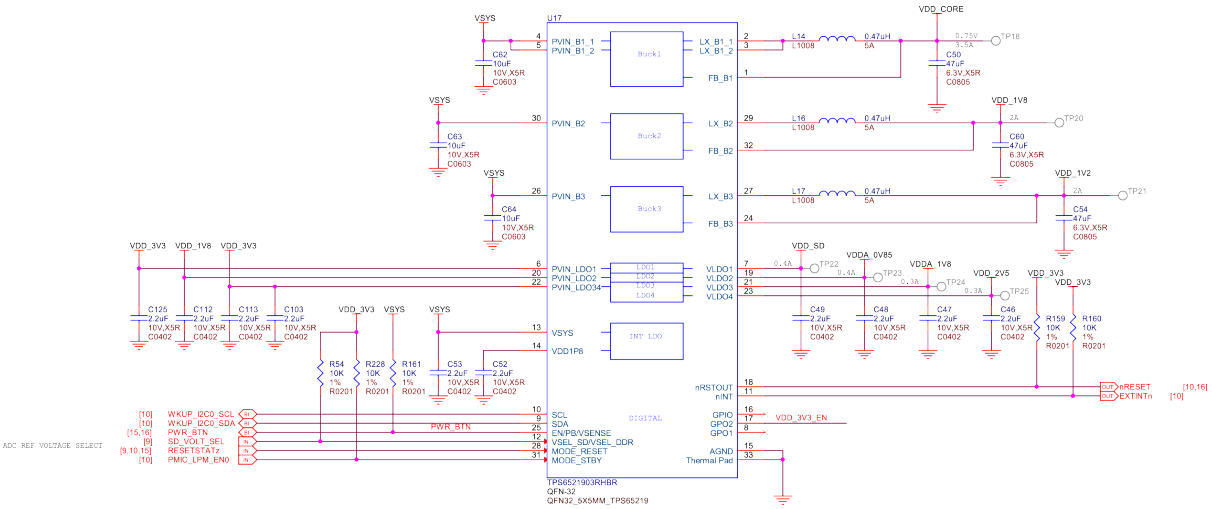


Fig. 3.22: PMIC

3.3.2 3V3 power

The TLV62595 is a high-efficiency, synchronous step-down converter that provides a stable 3.3V power supply to various components on PocketBeagle 2. It features a wide input voltage range, low quiescent current, and excellent transient response, making it suitable for powering sensitive electronics and ensuring reliable operation.

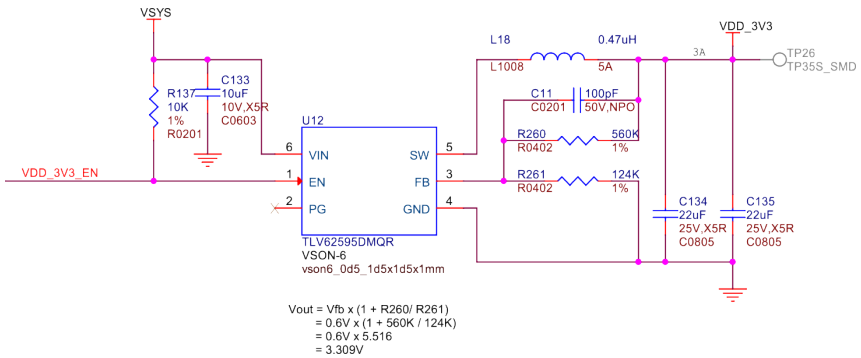


Fig. 3.23: 3V3 power

3.3.3 Power path

The LM73100 is a power path management IC that can be used to seamlessly switch between multiple power sources to generate a stable system voltage (VSYS). In this design, we have three power sources: VIN_5V, USB_5V, and VBAT. Here's how each of these sources is utilized:

1. VIN_5V: This is typically the main power input, which could come from an external power adapter. The LM73100 prioritizes this input when it is available, ensuring that the system is powered by this stable and higher current source.

2. USB_5V: This input comes from a USB connection. When VIN_5V is not available, the LM73100 switches to USB_5V to power the system. This allows the device to be powered or charged via a USB connection when an external adapter is not connected.

3. VBAT: This is the battery voltage input. When neither VIN_5V nor USB_5V is available, the LM73100 switches to VBAT to ensure that the system remains powered. This is crucial for portable devices that need to operate on battery power when no external power sources are connected.

The LM73100 manages these inputs and switches between them to provide a stable VSYS output. It ensures that the highest priority power source is used first, and seamlessly transitions to the next available source if the current one is disconnected or falls below a certain threshold.

This power path management ensures that the system remains powered without interruption, providing a reliable and efficient power solution for various applications.

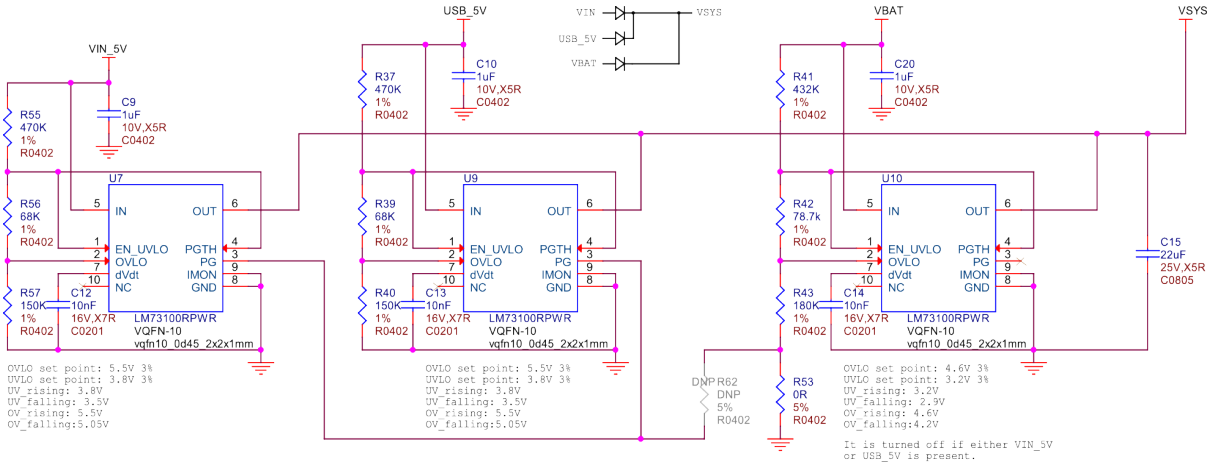


Fig. 3.24: Power path

3.3.4 Battery charging

The BQ21040 is a highly integrated Li-Ion and Li-Polymer linear battery charger device targeted at space-limited portable applications. The device operates from USB VBUS or cape header pin VIN voltage input. It features a high-accuracy voltage regulation, programmable charge current, and thermal regulation. The BQ21040 is designed to charge single-cell Li-Ion and Li-Polymer batteries and includes a power path management feature to power the system while charging the battery.

Key Features: - Input voltage range: 4.5V minimum - Programmable charge current up to 800mA - High-accuracy voltage regulation - Thermal regulation and protection - Power path management - Status indication for charge and fault conditions

Applications: - Wearable devices - Fitness accessories - Portable medical devices - Bluetooth headsets - Other space-limited portable applications

On PocketBeagle 2, the BQ21040 is used to manage the charging of a single-cell Li-Ion or Li-Polymer battery. The BQ21040's status indication feature provides feedback on the charging status and any fault conditions, making it easier to monitor the charging process. This integration of the BQ21040 in PocketBeagle 2 design enhances the device's portability and reliability, making it suitable for various applications that require battery power.

3.3.5 Decoupling capacitors

3.4 General connectivity and expansion

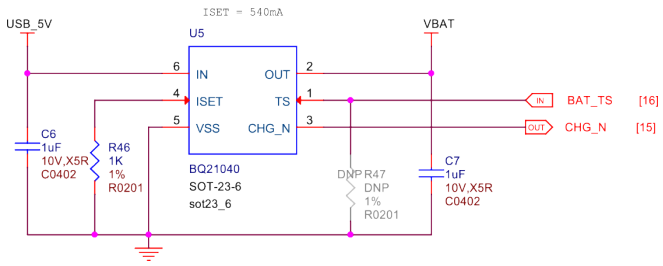


Fig. 3.25: Battery charging

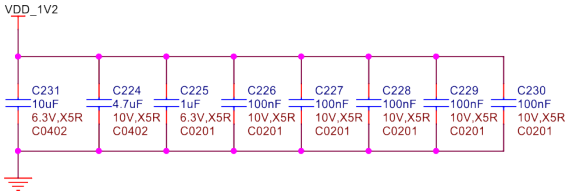


Fig. 3.26: VDD 1.2V capacitors

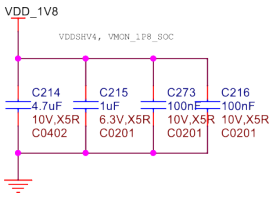


Fig. 3.27: VDD 1.8V capacitors

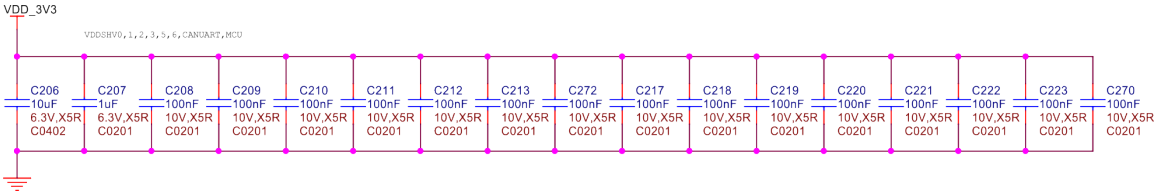


Fig. 3.28: VDD 3.3V capacitors

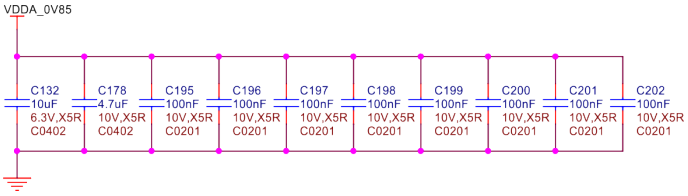


Fig. 3.29: VDDA 0.85V capacitors

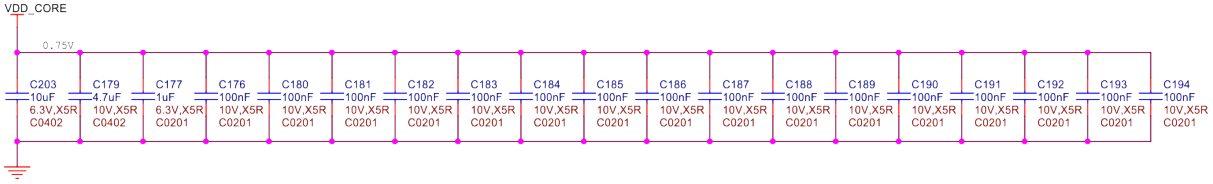


Fig. 3.30: VDD core capacitors

3.4.1 USB connections

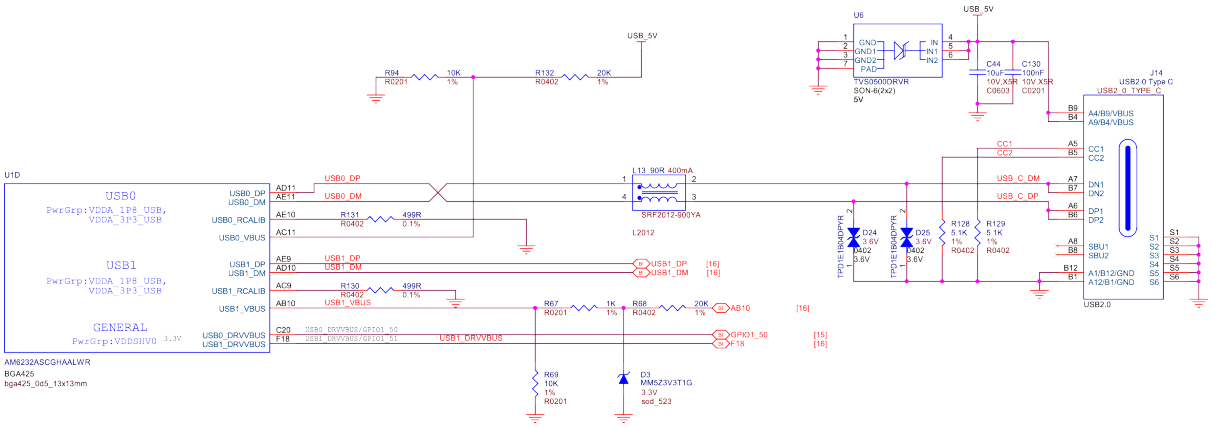


Fig. 3.31: USB connections

3.4.2 Cape headers

P1 cape header

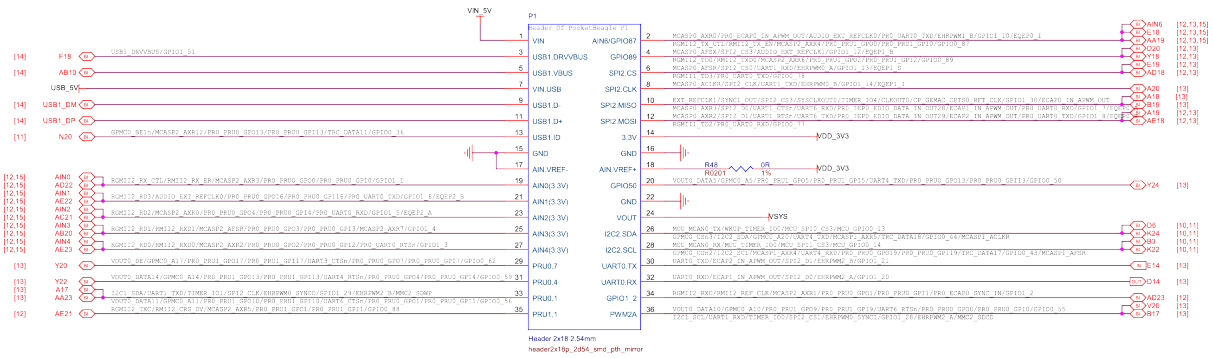


Fig. 3.32: P1 cape headers

P2 cape header

3.4.3 MicroSD card slot

3.5 Buttons & LEDs

3.5.1 User & Power Button

3.5.2 LED Indicators

3.6 Memory, Media, and storage

Described in the following sections are the memory devices found on the board.

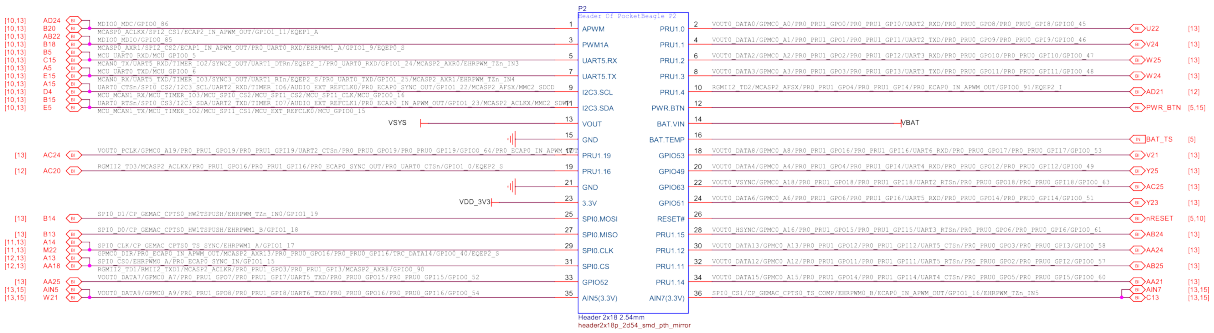


Fig. 3.33: P2 cape headers

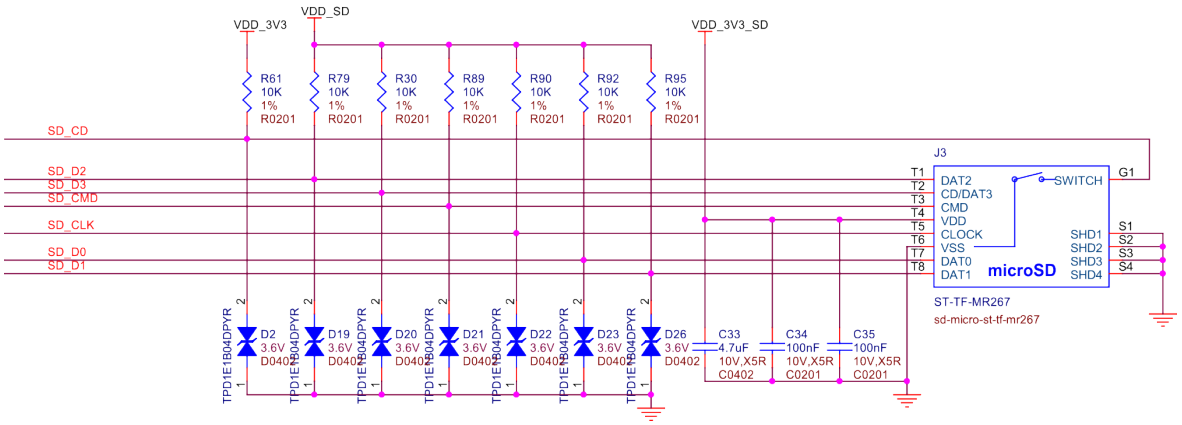


Fig. 3.34: MicroSD card slot

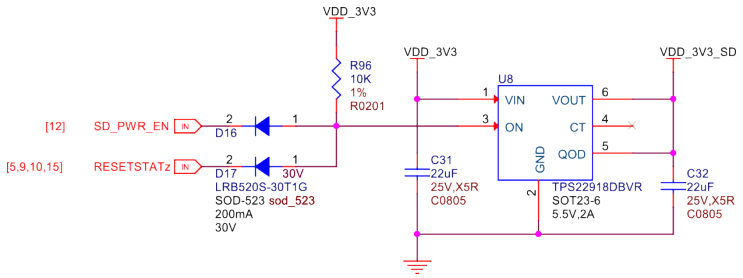


Fig. 3.35: MicroSD card power

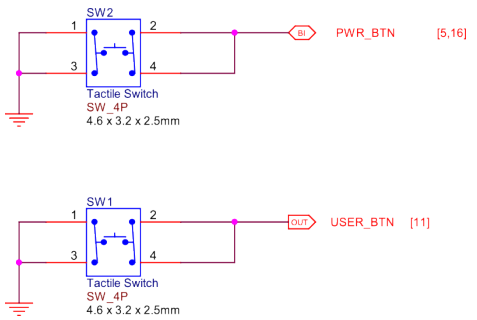


Fig. 3.36: Buttons

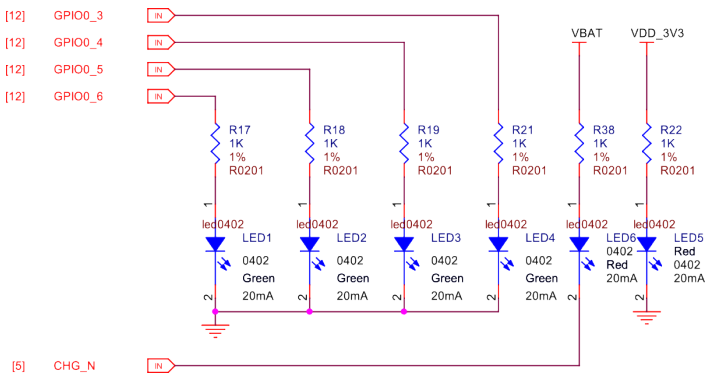


Fig. 3.37: LED indicators

3.6.1 512MB LPDDR4

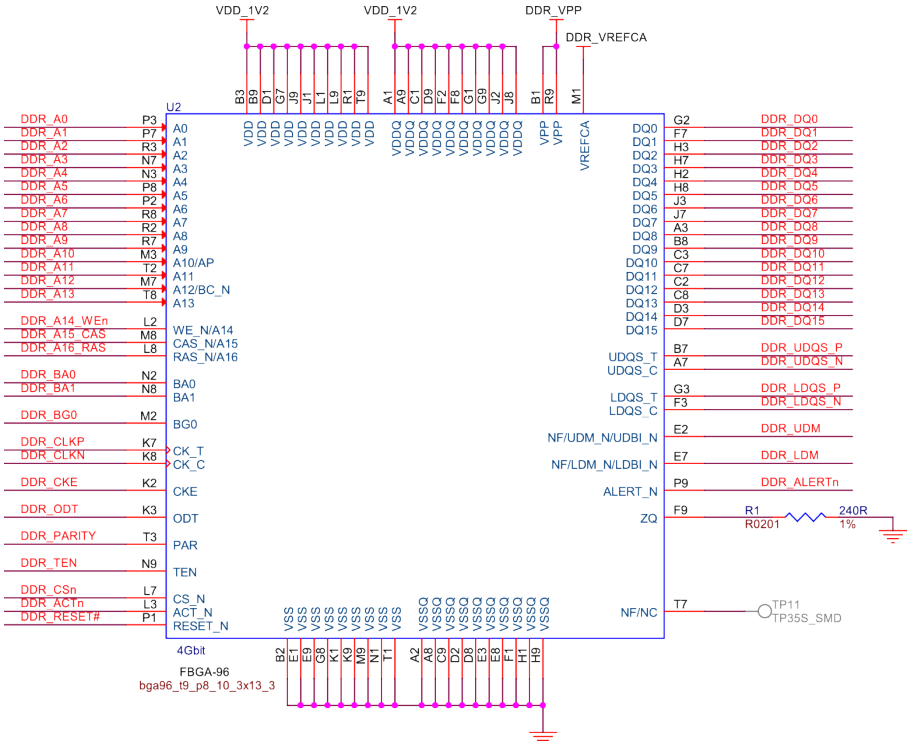


Fig. 3.38: 512MB LPDDR4 RAM

3.6.2 MSPM0 ADC & EEPROM

The MSPM0L1105 is a versatile microcontroller that we are utilizing to emulate an 8-channel 12-bit ADC and a 4KB EEPROM. This microcontroller is connected to PocketBeagle via the I2C interface, allowing for efficient communication and data transfer.

1. The 8-channel 12-bit ADC provides high-resolution analog-to-digital conversion, enabling precise measurement of analog signals from various sensors and inputs. This is particularly useful for applications requiring accurate data acquisition and monitoring. 1. The 4KB EEPROM emulation offers non-volatile storage for configuration data, calibration parameters, and other critical information. This ensures that important data is retained even when the system is powered off, enhancing the reliability and functionality of PocketBeagle 2.

By integrating the MSPM0L1105, we can leverage its capabilities to expand the analog input and storage options of PocketBeagle 2, making it suitable for a wider range of applications and use cases.

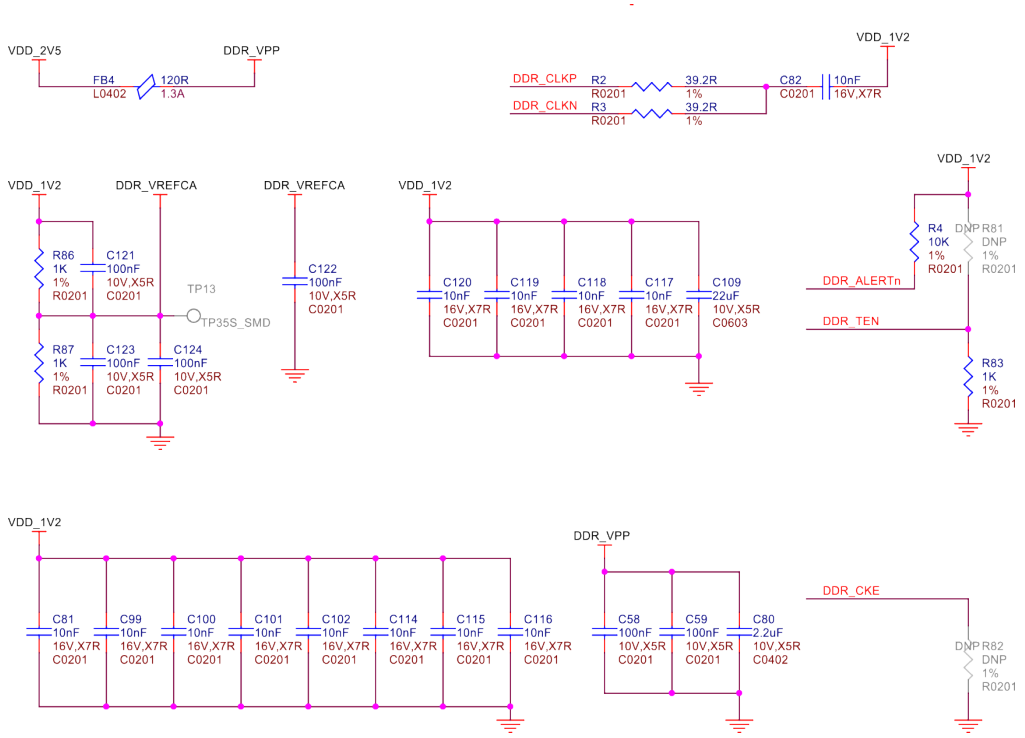


Fig. 3.39: DDR power

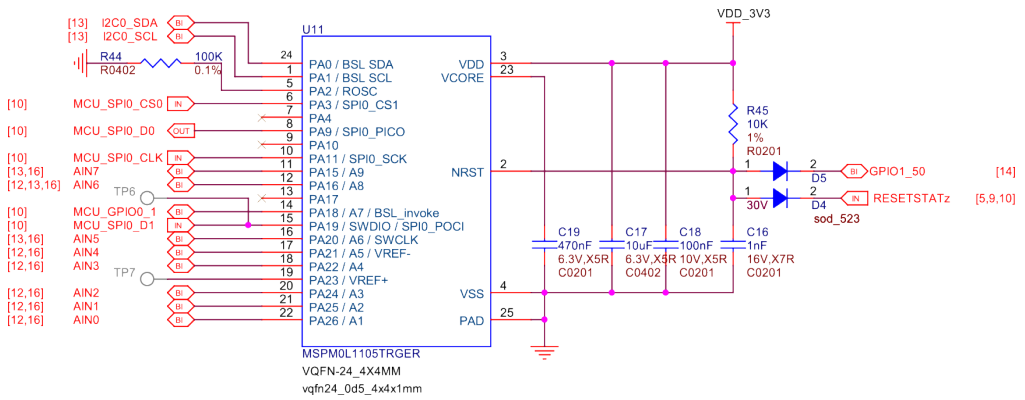


Fig. 3.40: MSPM0L1105 as 8ch 12bit ADC & 4KB EEPROM

3.7 Debug Ports

3.7.1 Serial debug port

PocketBeagle 2 features a JST-SH 1.00mm connector for UART, which is compatible with the Raspberry Pi Debug Probe. This connector allows for easy and reliable serial communication for boot time debugging purposes.

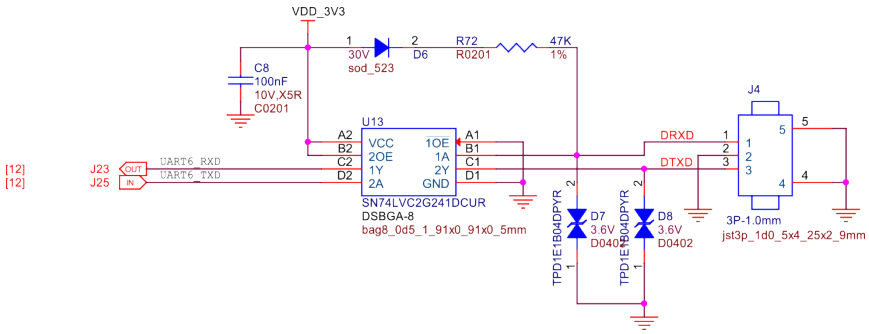


Fig. 3.41: Serial debug port

3.7.2 TagConnect (JTAG)

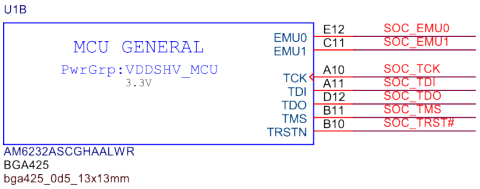


Fig. 3.42: JTAG

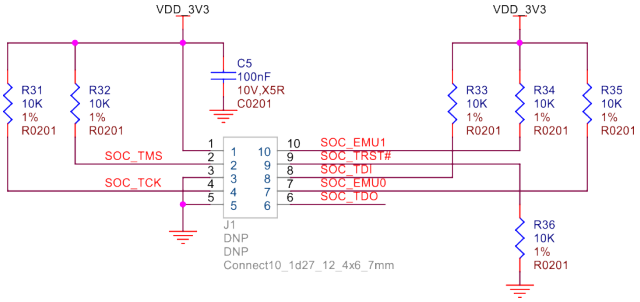


Fig. 3.43: TagConnect (JTAG)

3.8 Mechanical specifications

3.8.1 Dimensions & Weight

Table 3.2: Dimensions & weight

Parameter	Value
Size	56 x 35mm
Max heigh	13.6
PCB Size	55 x 35mm
PCB Layers	10–layers
PCB Thickness	1.6mm
RoHS compliant	Yes
Net Weight	12.7g
Gross Weight	19g

3.8.2 Board Dimensions

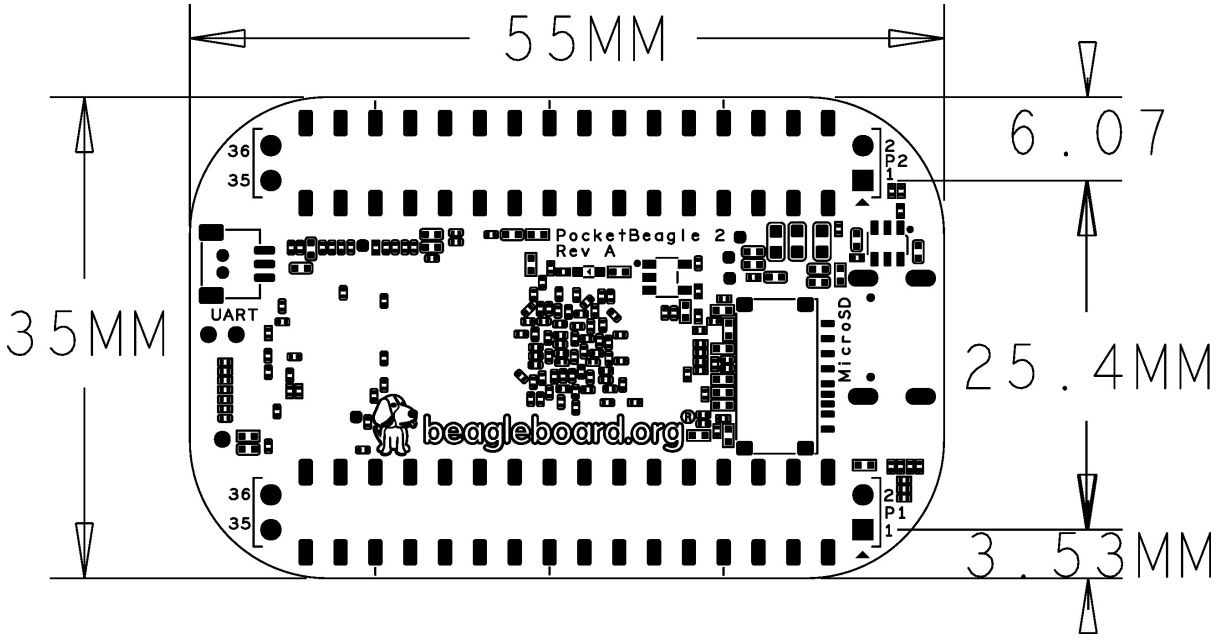


Fig. 3.44: PocketBeagle 2 RevA Dimensions

Chapter 4

Expansion

4.1 Pinout Diagrams

PocketBeagle 2 `P1` & `P2` cape headers are designed to be compatible with PocketBeagle classic as much as possible. Below pinout diagrams are design to simplify cape header pin usage and cape design process for PB2. To start using P1 / P2 cape header choose respective pinout diagram tab below.

P1 cape header

P2 cape header

4.2 Cape Header Connectors

Beagle cape expansion interface on PocketBeagle 2 like other Beagles is comprised of two headers P1 & P2. All signals on the expansion headers are **3.3V** unless otherwise indicated. **On some of the cape header pins on PocketBeagle 2 multiple SoC pins are shorted and only one of them should be used at a time.** Information regarding the double/shorted pins is provided in the *Pinout Diagrams* above (simplified) and cape header pin tables below (detailed).

> **Danger:** Do not connect 5V logic level signals to these pins or the board will be damaged.
>
> **NO PINS ARE TO BE DRIVEN UNTIL AFTER THE SYS_RESET LINE GOES HIGH. 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.**

4.2.1 Connector P1

The following tables show the pinout of the **P1** expansion header. The SW is responsible for setting the default function of each pin. 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 column heading is the pin number on the expansion header.

The **GPIO** row is the expected gpio identifier number in the Linux kernel.

Each row includes the gpiochipX and pinY in the format of *X Y*. You can use these values to directly control the GPIO pins with the commands shown below.

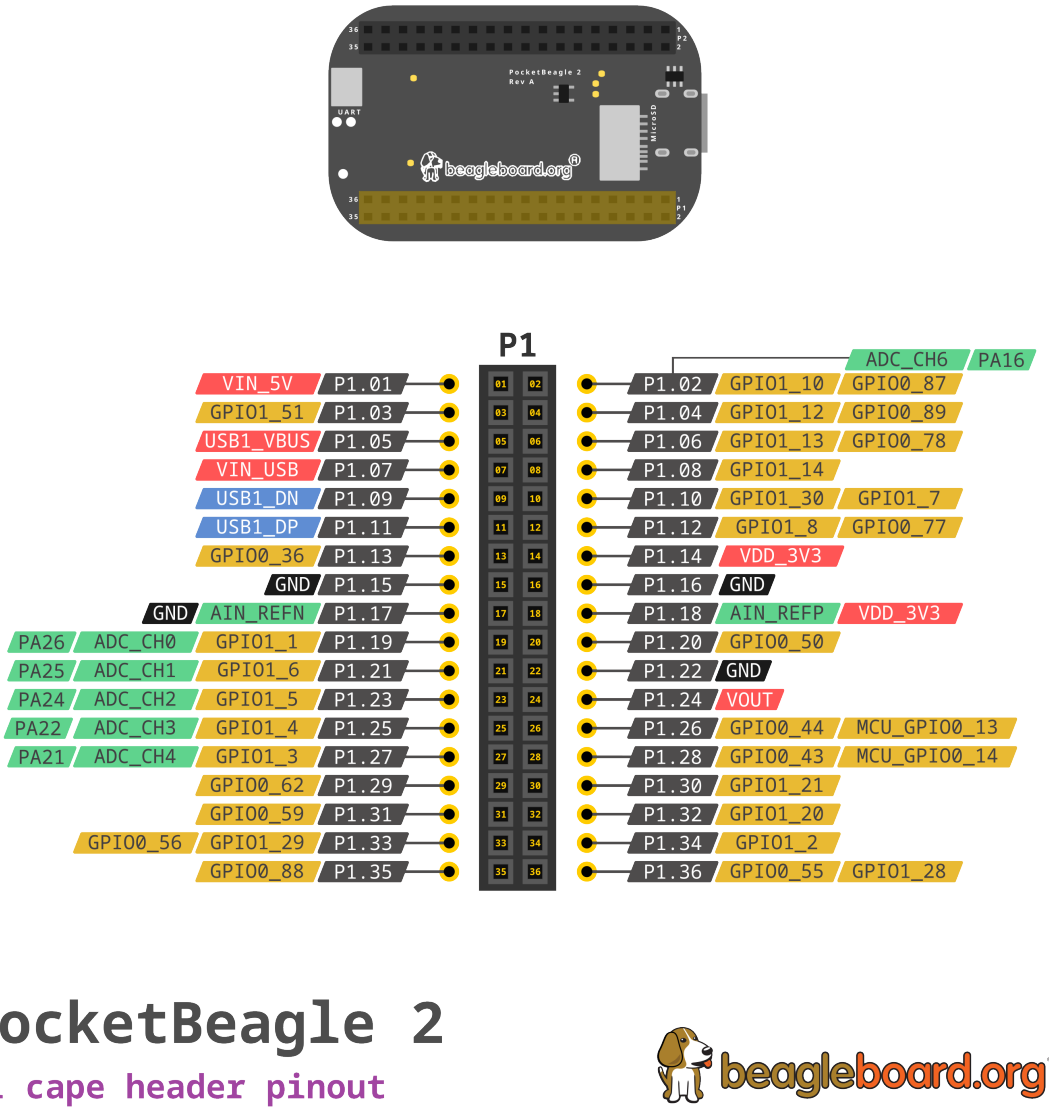


Fig. 4.1: PocketBeagle 2 P1 Cape Header Pinout

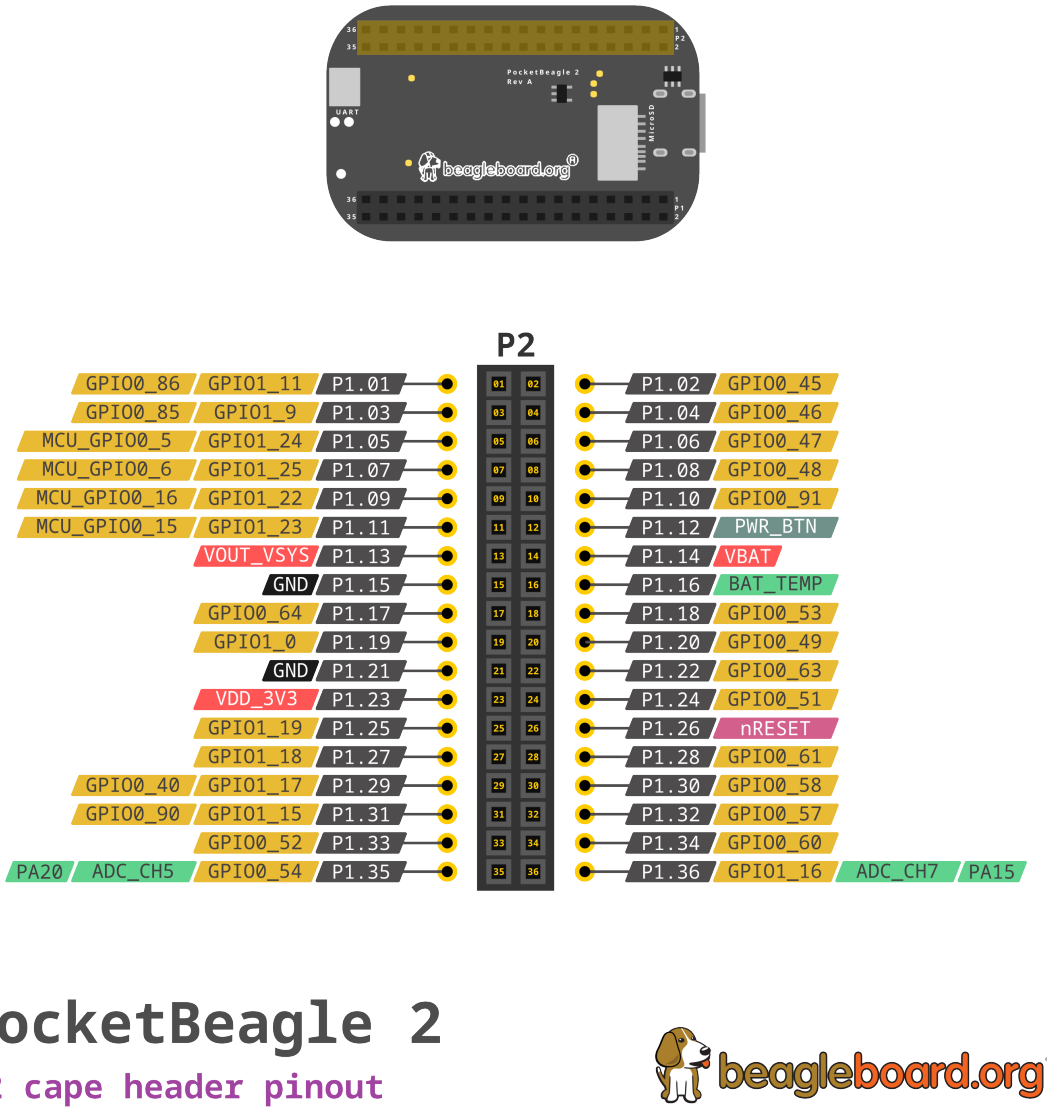


Fig. 4.2: PocketBeagle 2 P2 Cape Header Pinout

```
# to set the GPIO pin state to HIGH
debian@BeagleBone:~$ gpioset X Y=1

# to set the GPIO pin state to LOW
debian@BeagleBone:~$ gpioset X Y=0

For Example:

+---------+----------+
| Pin     | P1.03    |
+=========+==========+
| GPIO    | 1 20     |
+---------+----------+

Use the commands below for controlling this pin (P1.03) where X = 1 and Y =␣
↪20

# to set the GPIO pin state to HIGH
debian@BeagleBone:~$ gpioset 1 20=1

# to set the GPIO pin state to LOW
debian@BeagleBone:~$ gpioset 1 20=0
```

The **BALL** row is the pin number on the processor.

The **REG** row is the offset of the control register for the processor pin.

The **MODE #** rows are the mode setting for each pin. Setting each mode to align with the mode column will give that function on that pin.

Important: 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.

P1.02-P1.03

Pin	P1.01
Signal	VIN_5V

Pin	P1.02	P1.02A	P1.03
GPIO	GPIO1_10	GPIO0_87	GPIO1_51
BALL	E18	AA19	F18
Register	PADCONFIG104	PADCONFIG89	PADCONFIG150
Address	0x000F41A0	0x000F4164	0x000F4258
Page	29	40	45
MODE 0	MCASP0_AXR0	RGMII2_TX_CTL	USB1_DRVVBUS
MODE 1	PR0_ECAP0_IN_APWM_OUT	RMII2_TX_EN	~
MODE 2	AUDIO_EXT_REFCLK0	MCASP2_AXR4	~
MODE 3	~	PR0_PRU1_GPO0	~
MODE 4	~	PR0_PRU1_GPI0	~
MODE 5	PR0_UART0_TXD	~	~
MODE 6	EHRPWM1_B	~	~
MODE 7	GPIO1_10	GPIO0_87	GPIO1_51
MODE 8	EQEP0_I	~	~

P1.04-P1.07

Pin	P1.05	P1.07
BALL	AB10	~
Signal	USB1_VBUS	VIN.USB

Pin	P1.04	P1.04A	P1.06	P1.06A
GPIO	GPIO1_12	GPIO0_89	GPIO1_13	GPIO0_78
BALL	D20	Y18	E19	AD18
Register	PADCONFIG106	PADCONFIG91	PADCONFIG107	PADCONFIG80
Address	0x000F41A8	0x000F416C	0x000F41AC	0x000F4140
Page	29	42	29	41
MODE 0	MCASP0_AFSX	RGMII2_TD0	MCASP0_AFSR	RGMII1_TD3
MODE 1	SPI2_CS3	RMII2_TXD0	SPI2_CS0	~
MODE 2	AUDIO_EXT_REFCLK1	MCASP2_AXR6	UART1_RXD	PR0_UART0_TXD
MODE 3	~	PR0_PRU1_GPO2	~	~
MODE 4	~	PR0_PRU1_GPI2	~	~
MODE 5	~	~	~	~
MODE 6	~	~	EHRPWM0_A	~
MODE 7	GPIO1_12	GPIO0_89	GPIO1_13	GPIO0_78
MODE 8	EQEP1_B	~	EQEP1_S	~

P1.08 - P1.11

Pin	P1_09	P1_11
BALL	AD10	AE9
Signal	USB1.D-	USB1.D+

Pin	P1.08	P1.10	P1.10A
GPIO	GPIO1_14	GPIO1_30	GPIO1_7
BALL	A20	A18	B19
Register	PADCONFIG108	PADCONFIG124	PADCONFIG101
Address	0x000F41B0	0x000F41F0	0x000F4194
Page	28	19	30
MODE 0	MCASP0_ACLKR	EXT_REFCLK1	MCASP0_AXR3
MODE 1	SPI2_CLK	SYNC1_OUT	SPI2_D0
MODE 2	UART1_TXD	SPI2_CS3	UART1_CTSn
MODE 3	~	SYSCLKOUT0	UART6_RXD
MODE 4	~	TIMER_IO4	PR0_IEP0_EDIO_DATA_IN_OUT28
MODE 5	~	CLKOUT0	ECAP1_IN_APWM_OUT
MODE 6	EHRPWM0_B	CP_GEMAC_CPTS0_RFT_CLK	PR0_UART0_RXD
MODE 7	GPIO1_14	GPIO1_30	GPIO1_7
MODE 8	EQEP1_I	ECAP0_IN_APWM_OUT	EQEP0_A

P1.12 - P1.13

Pin	P1.12	P1.12A	P1.13
GPIO	GPIO1_8	GPIO0_77	GPIO0_36
BALL	A19	AE18	N20
Register	PADCONFIG102	PADCONFIG79	PADCONFIG37
Address	0x000F4198	0x000F413C	0x000F4094
Page	29	41	25
MODE 0	MCASP0_AXR2	RGMII1_TD2	GPMC0_BE1n
MODE 1	SPI2_D1	~	~
MODE 2	UART1_RTSn	PR0_UART0_RXD	~
MODE 3	UART6_TXD	~	MCASP2_AXR12
MODE 4	PR0_IEP0_EDIO_DATA_IN_OUT29	~	PR0_PRU0_GPO13
MODE 5	ECAP2_IN_APWM_OUT	~	PR0_PRU0_GPI13
MODE 6	PR0_UART0_TXD	~	TRC_DATA11
MODE 7	GPIO1_8	GPIO0_77	GPIO0_36
MODE 8	EQEP0_B	~	~

P1.14 - P1.21

Pin	P1_14	P1_15	P1_16	P1_17	P1_18	P1_19	P1_21
Signal	VDD_3V3	GND	GND	AIN.REF-	AIN.REF+	AIN0	AIN1

Pin	P1.19	P1.20	P1.21
GPIO	GPIO1_1	GPIO0_50	GPIO1_6
BALL	AD22	Y24	AE22
Register	PADCONFIG95	PADCONFIG51	PADCONFIG100
Address	0x000F417C	0x000F40CC	0x000F4190
Page	40	49	42
MODE 0	RGMII2_RX_CTL	VOUT0_DATA5	RGMII2_RD3
MODE 1	RMII2_RX_ER	GPMC0_A5	~
MODE 2	MCASP2_AXR3	PR0_PRU1_GPO5	AUDIO_EXT_REFCLK0
MODE 3	PR0_PRU0_GPO0	PR0_PRU1_GPI5	PR0_PRU0_GPO16
MODE 4	PR0_PRU0_GPI0	UART4_TXD	PR0_PRU0_GPI16
MODE 5	~	PR0_PRU0_GPO13	PR0_UART0_TXD
MODE 6	~	PR0_PRU0_GPI13	~
MODE 7	GPIO1_1	GPIO0_50	GPIO1_6
MODE 8	~	~	EQEP2_B

P1.22 - P1.26

Pin	P1.22	P1.23	P1.24	P1.25
Signal	GND	AIN2	VOUT	AIN3

Pin	P1.23	P1.25	P1.26	P1.26A
GPIO	GPIO1_5	GPIO1_4	GPIO0_44	MCU_GPIO0_13
BALL	AC21	AB20	K24	D6
Register	PADCONFIG99	PADCONFIG98	PADCONFIG45	MCU_PADCONFIG13
Address	0x000F418C	0x000F4188	0x000F40B4	0x04084034
Page	41	41	26	30
MODE 0	RGMII2_RD2	RGMII2_RD1	GPMC0_CSn3	MCU_MCAN0_TX
MODE 1	~	RMII2_RXD1	I2C2_SDA	WKUP_TIMER_IO0
MODE 2	MCASP2_AXR0	MCASP2_AFSR	GPMC0_A20	MCU_SPI0_CS3
MODE 3	PR0_PRU0_GPO4	PR0_PRU0_GPO3	UART4_TXD	~
MODE 4	PR0_PRU0_GPI4	PR0_PRU0_GPI3	MCASP1_AXR5	~
MODE 5	PR0_UART0_RXD	MCASP2_AXR7	~	~
MODE 6	~	~	TRC_DATA18	~
MODE 7	GPIO1_5	GPIO1_4	GPIO0_44	MCU_GPIO0_13
MODE 8	EQEP2_A	~	MCASP1_ACLKR	~

P1.27 - P1.28

Pin	P1.27
Signal	AIN4

Pin	P1.27	P1.28	P1.28A
GPIO	GPIO1_3	GPIO0_43	MCU_GPIO0_14
BALL	AE23	K22	B3
Register	PADCONFIG97	PADCONFIG44	MCU_PADCONFIG14
Address	0x000F4184	0x000F40B0	0x04084038
Page	41	49	43
MODE 0	RGMII2_RD0	GPMC0_CSn2	MCU_MCAN0_RX
MODE 1	RMII2_RXD0	I2C2_SCL	MCU_TIMER_IO0
MODE 2	MCASP2_AXR2	MCASP1_AXR4	MCU_SPI1_CS3
MODE 3	PR0_PRU0_GPO2	UART4_RXD	~
MODE 4	PR0_PRU0_GPI2	PR0_PRU0_GPO19	~
MODE 5	~	PR0_PRU0_GPI19	~
MODE 6	PR0_UART0_RTSn	TRC_DATA17	~
MODE 7	GPIO1_3	GPIO0_43	MCU_GPIO0_14
MODE 8	~	MCASP1_AFSR	~

P1.29 - P1.31

Pin	P1.29	P1.30	P1.31
GPIO	GPIO0_62	GPIO1_21	GPIO0_59
BALL	Y20	E14	Y22
Register	PADCONFIG63	PADCONFIG115	PADCONFIG60
Address	0x000F40FC	0x000F41CC	0x000F40F0
Page	46	45	51
MODE 0	VOUT0_DE	UART0_TXD	VOUT0_DATA14
MODE 1	GPMC0_A17	ECAP2_IN_APWM_OUT	GPMC0_A14
MODE 2	PR0_PRU1_GPO17	SPI2_D1	PR0_PRU1_GPO13
MODE 3	PR0_PRU1_GPI17	EHRPWM2_B	PR0_PRU1_GPI13
MODE 4	UART3_CTSn	~	UART4_RTSn
MODE 5	PR0_PRU0_GPO7	~	PR0_PRU0_GPO4
MODE 6	PR0_PRU0_GPI7	~	PR0_PRU0_GPI4
MODE 7	GPIO0_62	GPIO1_21	GPIO0_59
MODE 8	~	~	~

P1.32 - P1.33

Pin	P1.32	P1.33	P1.33A
GPIO	GPIO1_20	GPIO1_29	GPIO0_56
BALL	D14	A17	AA23
Register	PADCONFIG114	PADCONFIG123	PADCONFIG57
Address	0x000F41C8	0x000F41EC	0x000F40E4
Page	44	27	50
MODE 0	UART0_RXD	I2C1_SDA	VOUT0_DATA11
MODE 1	ECAP1_IN_APWM_OUT	UART1_TXD	GPMC0_A11
MODE 2	SPI2_D0	TIMER_IO1	PR0_PRU1_GPO10
MODE 3	EHRPWM2_A	SPI2_CLK	PR0_PRU1_GPI10
MODE 4	~	EHRPWM0_SYNCO	UART6_CTSn
MODE 5	~	~	PR0_PRU0_GPO1
MODE 6	~	~	PR0_PRU0_GPI1
MODE 7	GPIO1_20	GPIO1_29	GPIO0_56
MODE 8	~	EHRPWM2_B	~
MODE 9	~	MMC2_SDWP	~

P1.34 - P1.36

Pin	P1.34	P1.35	P1.36	P1.36A
GPIO	GPIO1_2	GPIO0_88	GPIO0_55	GPIO1_28
BALL	AD23	AE21	V20	B17
Register	PADCONFIG96	PADCONFIG90	PADCONFIG56	PADCONFIG122
Address	0x000F4180	0x000F4168	0x000F40E0	0x000F41E8
Page	40	40	50	27
MODE 0	RGMII2_RXC	RGMII2_TXC	VOUT0_DATA10	I2C1_SCL
MODE 1	RMII2_REF_CLK	RMII2_CRS_DV	GPMC0_A10	UART1_RXD
MODE 2	MCASP2_AXR1	MCASP2_AXR5	PR0_PRU1_GPO9	TIMER_IO0
MODE 3	PR0_PRU0_GPO1	PR0_PRU1_GPO1	PR0_PRU1_GPI9	SPI2_CS1
MODE 4	PR0_PRU0_GPI1	PR0_PRU1_GPI1	UART6_RTSn	EHRPWM0_SYNCI
MODE 5	PR0_ECAP0_SYNC_IN	~	PR0_PRU0_GPO0	~
MODE 6	~	~	PR0_PRU0_GPI0	~
MODE 7	GPIO1_2	GPIO0_88	GPIO0_55	GPIO1_28
MODE 8	~	~	~	EHRPWM2_A
MODE 9	~	~	~	MMC2_SDCD

4.2.2 Connector P2

The following tables show the pinout of the **P2** expansion header. The SW is responsible for setting the default function of each pin. 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 column heading is the pin number on the expansion header.

The **GPIO** row is the expected gpio identifier number in the Linux kernel.

Each row includes the gpiochipX and pinY in the format of *X Y*. You can use these values to directly control the GPIO pins with the commands shown below.

```
# to set the GPIO pin state to HIGH
debian@BeagleBone:~$ gpioset X Y=1

# to set the GPIO pin state to LOW
debian@BeagleBone:~$ gpioset X Y=0

For Example:

+---------+----------+
```

(continues on next page)

(continued from previous page)

```
| Pin      | P2.11     |
+==========+===========+
| GPIO     | 1 1       |
+----------+-----------+

Use the commands below for controlling this pin (P2.11) where X = 1 and Y = 1

# to set the GPIO pin state to HIGH
debian@BeagleBone:~$ gpioset 1 20=1

# to set the GPIO pin state to LOW
debian@BeagleBone:~$ gpioset 1 20=0
```

The **BALL** row is the pin number on the processor.

The **REG** row is the offset of the control register for the processor pin.

The **MODE #** rows are the mode setting for each pin. Setting each mode to align with the mode column will give that function on that pin.

If included, the **2nd BALL** row is the pin number on the processor for a second processor pin connected to the same pin on the expansion header. Similarly, all row headings starting with **2nd** refer to data for this second processor pin.

Important: 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.

P2.01 - P2.02

Pin	P2.01	P2.01A	P2.02
GPIO	GPIO1_11	GPIO0_86	GPIO0_45
BALL	B20	AD24	U22
Register	PADCONFIG105	PADCONFIG88	PADCONFIG46
Address	0x000F41A4	0x000F4160	0x000F40B8
Page	28	32	47
MODE 0	MCASP0_ACLKX	MDIO0_MDC	VOUT0_DATA0
MODE 1	SPI2_CS1	~	GPMC0_A0
MODE 2	ECAP2_IN_APWM_OUT	~	PR0_PRU1_GPO0
MODE 3	~	~	PR0_PRU1_GPI0
MODE 4	~	~	UART2_RXD
MODE 5	~	~	PR0_PRU0_GPO8
MODE 6	~	~	PR0_PRU0_GPI8
MODE 7	GPIO1_11	GPIO0_86	GPIO0_45
MODE 8	EQEP1_A	~	~

P2.03 - P2.04

Pin	P2.03A	P2.03	P2.04
GPIO	GPIO1_9	GPIO0_85	GPIO0_46
BALL	B18	AB22	V24
Register	PADCONFIG103	PADCONFIG87	PADCONFIG47
Address	0x000F419C	0x000F415C	0x000F40BC
Page	29	32	48
MODE 0	MCASP0_AXR1	MDIO0_MDIO	VOUT0_DATA1
MODE 1	SPI2_CS2	~	GPMC0_A1
MODE 2	ECAP1_IN_APWM_OUT	~	PR0_PRU1_GPO1
MODE 3	~	~	PR0_PRU1_GPI1
MODE 4	~	~	UART2_TXD
MODE 5	PR0_UART0_RXD	~	PR0_PRU0_GPO9
MODE 6	EHRPWM1_A	~	PR0_PRU0_GPI9
MODE 7	GPIO1_9	GPIO0_85	GPIO0_46
MODE 8	EQEP0_S	~	~

P2.05 - P2.06

Pin	P2.05	P2.05A	P2.06
GPIO	GPIO1_24	MCU_GPIO0_5	GPIO0_47
BALL	C15	B5	W25
Register	PADCONFIG118	MCU_PADCONFIG5	PADCONFIG48
Address	0x000F41D8	0x04084014	0x000F40C0
Page	28	32	48
MODE 0	MCAN0_TX	MCU_UART0_RXD	VOUT0_DATA2
MODE 1	UART5_RXD	~	GPMC0_A2
MODE 2	TIMER_IO2	~	PR0_PRU1_GPO2
MODE 3	SYNC2_OUT	~	PR0_PRU1_GPI2
MODE 4	UART1_DTRn	~	UART3_RXD
MODE 5	EQEP2_I	~	PR0_PRU0_GPO10
MODE 6	PR0_UART0_RXD	~	PR0_PRU0_GPI10
MODE 7	GPIO1_24	MCU_GPIO0_5	GPIO0_47
MODE 8	MCASP2_AXR0	~	~
MODE 9	EHRPWM_TZn_IN3	~	~

P2.07 - P2.08

Pin	P2.07	P2.07A	P2.08
GPIO	GPIO1_25	MCU_GPIO0_6	GPIO0_48
BALL	E15	A5	W24
Register	PADCONFIG119	MCU_PADCONFIG6	PADCONFIG49
Address	0x000F41DC	0x04084018	0x000F40C4
Page	28	32	48
MODE 0	MCAN0_RX	MCU_UART0_TXD	VOUT0_DATA3
MODE 1	UART5_TXD	~	GPMC0_A3
MODE 2	TIMER_IO3	~	PR0_PRU1_GPO3
MODE 3	SYNC3_OUT	~	PR0_PRU1_GPI3
MODE 4	UART1_RIn	~	UART3_TXD
MODE 5	EQEP2_S	~	PR0_PRU0_GPO11
MODE 6	PR0_UART0_TXD	~	PR0_PRU0_GPI11
MODE 7	GPIO1_25	MCU_GPIO0_6	GPIO0_48
MODE 8	MCASP2_AXR1	~	~
MODE 9	EHRPWM_TZn_IN4	~	~

P2.09 - P2.10

Pin	P2.09	P2.09A	P2.10
GPIO	GPIO1_22	MCU_GPIO0_16	GPIO0_91
BALL	A15	D4	AD21
Register	PADCONFIG116	MCU_PADCONFIG16	PADCONFIG93
Address	0x000F41D0	0x04084040	0x000F4174
Page	44	30	42
MODE 0	UART0_CTSn	MCU_MCAN1_RX	RGMII2_TD2
MODE 1	SPI0_CS2	MCU_TIMER_IO3	~
MODE 2	I2C3_SCL	MCU_SPI0_CS2	MCASP2_AFSX
MODE 3	UART2_RXD	MCU_SPI1_CS2	PR0_PRU1_GPO4
MODE 4	TIMER_IO6	MCU_SPI1_CLK	PR0_PRU1_GPI4
MODE 5	AUDIO_EXT_REFCLK0	~	PR0_ECAP0_IN_APWM_OUT
MODE 6	PR0_ECAP0_SYNC_OUT	~	~
MODE 7	GPIO1_22	MCU_GPIO0_16	GPIO0_91
MODE 8	MCASP2_AFSX	~	EQEP2_I
MODE 9	MMC2_SDCD	~	~

P2.11 - P2.17

Pin	P2.12	P2.13	P2.14	P2.15	P2.16
Signal	PWR.BTN	VOUT - VSYS	VBAT	GND	BAT.TEMP - BAT_TS

Pin	P2.11	P2.11A	P2.17
GPIO	GPIO1_23	MCU_GPIO0_15	GPIO0_64
BALL	B15	E5	AC24
Register	PADCONFIG117	MCU_PADCONFIG15	PADCONFIG65
Address	0x000F41D4	0x0408403C	0x000F4104
Page	44	31	47
MODE 0	UART0_RTSn	MCU_MCAN1_TX	VOUT0_PCLK
MODE 1	SPI0_CS3	MCU_TIMER_IO2	GPMC0_A19
MODE 2	I2C3_SDA	~	PR0_PRU1_GPO19
MODE 3	UART2_TXD	MCU_SPI1_CS1	PR0_PRU1_GPI19
MODE 4	TIMER_IO7	MCU_EXT_REFCLK0	UART2_CTSn
MODE 5	AUDIO_EXT_REFCLK1	~	PR0_PRU0_GPO19
MODE 6	PR0_ECAP0_IN_APWM_OUT	~	PR0_PRU0_GPI19
MODE 7	GPIO1_23	MCU_GPIO0_15	GPIO0_64
MODE 8	MCASP2_ACLKX	~	PR0_ECAP0_IN_APWM_OUT
MODE 9	MMC2_SDWP	~	~

P2.18 - P2.23

Pin	P1.21	P2.23
Signal	GND	VDD_3V3

Pin	P2.18	P2.19	P2.20	P2.22
GPIO	GPIO0_53	GPIO1_0	GPIO0_49	GPIO0_63
BALL	V21	AC20	Y25	AC25
Register	PADCONFIG54	PADCONFIG94	PADCONFIG50	PADCONFIG64
Address	0x000F40D8	0x000F4178	0x000F40C8	0x000F4100
Page	49	42	48	47
MODE 0	VOUT0_DATA8	RGMII2_TD3	VOUT0_DATA4	VOUT0_VSYNC
MODE 1	GPMC0_A8	~	GPMC0_A4	GPMC0_A18
MODE 2	PR0_PRU1_GPO16	MCASP2_ACLKX	PR0_PRU1_GPO4	PR0_PRU1_GPO18
MODE 3	PR0_PRU1_GPI16	PR0_PRU1_GPO16	PR0_PRU1_GPI4	PR0_PRU1_GPI18
MODE 4	UART6_RXD	PR0_PRU1_GPI16	UART4_RXD	UART2_RTSn
MODE 5	PR0_PRU0_GPO17	PR0_ECAP0_SYNC_OUT	PR0_PRU0_GPO12	PR0_PRU0_GPO18
MODE 6	PR0_PRU0_GPI17	PR0_UART0_CTSn	PR0_PRU0_GPI12	PR0_PRU0_GPI18
MODE 7	GPIO0_53	GPIO1_0	GPIO0_49	GPIO0_63
MODE 8	~	EQEP2_S	~	~

P2.24 - P2.27

Pin	P1.26
Signal	RESET# - nRESET

Pin	P2.24	P2.25	P2.27
GPIO	GPIO0_51	GPIO1_19	GPIO1_18
BALL	Y23	B14	B13
Register	PADCONFIG52	PADCONFIG113	PADCONFIG112
Address	0x000F40D0	0x000F41C4	0x000F41C0
Page	49	43	43
MODE 0	VOUT0_DATA6	SPI0_D1	SPI0_D0
MODE 1	GPMC0_A6	CP_GEMAC_CPTS0_HW2TSPUSH	CP_GEMAC_CPTS0_HW1TSPUSH
MODE 2	PR0_PRU1_GPO6	EHRPWM_TZn_IN0	EHRPWM1_B
MODE 3	PR0_PRU1_GPI6	~	~
MODE 4	UART5_RXD	~	~
MODE 5	PR0_PRU0_GPO14	~	~
MODE 6	PR0_PRU0_GPI14	~	~
MODE 7	GPIO0_51	GPIO1_19	GPIO1_18
MODE 8	~	~	~

P2.28 - P2.39

Pin	P2.28	P2.29	P2.29A
GPIO	GPIO0_61	GPIO1_17	GPIO0_40
BALL	AB24	A14	M22
Register	PADCONFIG62	PADCONFIG111	PADCONFIG41
Address	0x000F40F8	0x000F41BC	0x000F40A4
Page	47	43	20
MODE 0	VOUT0_HSYNC	SPI0_CLK	GPMC0_DIR
MODE 1	GPMC0_A16	CP_GEMAC_CPTS0_TS_SYNC	PR0_ECAP0_IN_APWM_OUT
MODE 2	PR0_PRU1_GPO15	EHRPWM1_A	~
MODE 3	PR0_PRU1_GPI15	~	MCASP2_AXR13
MODE 4	UART3_RTSn	~	PR0_PRU0_GPO16
MODE 5	PR0_PRU0_GPO6	~	PR0_PRU0_GPI16
MODE 6	PR0_PRU0_GPI6	~	TRC_DATA14
MODE 7	GPIO0_61	GPIO1_17	GPIO0_40
MODE 8	~	~	EQEP2_S

P2.30 - P2.31

Pin	P2.30	P2.31	P2.31A
GPIO	GPIO0_58	GPIO1_15	GPIO0_90
BALL	AA24	A13	AA18
Register	PADCONFIG59	PADCONFIG109	PADCONFIG92
Address	0x000F40EC	0x000F41B4	0x000F4170
Page	51	42	42
MODE 0	VOUT0_DATA13	SPI0_CS0	RGMII2_TD1
MODE 1	GPMC0_A13	~	RMII2_TXD1
MODE 2	PR0_PRU1_GPO12	EHRPWM0_A	MCASP2_ACLKR
MODE 3	PR0_PRU1_GPI12	~	PR0_PRU1_GPO3
MODE 4	UART5_CTSn	~	PR0_PRU1_GPI3
MODE 5	PR0_PRU0_GPO3	~	MCASP2_AXR8
MODE 6	PR0_PRU0_GPI3	PR0_ECAP0_SYNC_IN	~
MODE 7	GPIO0_58	GPIO1_15	GPIO0_90

P2.32 - P2.34

Pin	P2.32	P2.33	P2.34
GPIO	GPIO0_57	GPIO0_52	GPIO0_60
BALL	AB25	AA25	AA21
Register	PADCONFIG58	PADCONFIG53	PADCONFIG61
Address	0x000F40E8	0x000F40D4	0x000F40F4
Page	50	49	51
MODE 0	VOUT0_DATA12	VOUT0_DATA7	VOUT0_DATA15
MODE 1	GPMC0_A12	GPMC0_A7	GPMC0_A15
MODE 2	PR0_PRU1_GPO11	PR0_PRU1_GPO7	PR0_PRU1_GPO14
MODE 3	PR0_PRU1_GPI11	PR0_PRU1_GPI7	PR0_PRU1_GPI14
MODE 4	UART5_RTSn	UART5_TXD	UART4_CTSn
MODE 5	PR0_PRU0_GPO2	PR0_PRU0_GPO15	PR0_PRU0_GPO5
MODE 6	PR0_PRU0_GPI2	PR0_PRU0_GPI15	PR0_PRU0_GPI5
MODE 7	GPIO0_57	GPIO0_52	GPIO0_60

P2.35 - P2.36

Pin	P2.35	P2.36
Signal	AIN5	AIN7

Pin	P2.35	P2.36
GPIO	GPIO0_54	GPIO1_16
BALL	W21	C13
Register	PADCONFIG55	PADCONFIG110
Address	0x000F40DC	0x000F41B8
Page	50	43
MODE 0	VOUT0_DATA9	SPI0_CS1
MODE 1	GPMC0_A9	CP_GEMAC_CPTS0_TS_COMP
MODE 2	PR0_PRU1_GPO8	EHRPWM0_B
MODE 3	PR0_PRU1_GPI8	ECAP0_IN_APWM_OUT
MODE 4	UART6_TXD	~
MODE 5	PR0_PRU0_GPO16	~
MODE 6	PR0_PRU0_GPI16	~
MODE 7	GPIO0_54	GPIO1_16
MODE 8	~	~
MODE 9	~	EHRPWM_TZn_IN5

Chapter 5

Demos and Tutorials

Chapter 6

Additional Support Information

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

6.1 Certifications and export control

6.1.1 Export designations

- ECCN: 3A991.a
- HSCODE: 8517180050
- USHSCODE: 8543708800
- EUHSCODE: 8543709099
- UPC number: 841454123484

6.2 Hardware Design

You can find all PocketBeagle 2 hardware files here under the *design* folder.

6.3 Production board boot media

6.4 Software Updates

Follow instructions below to download the latest image for your PocketBeagle 2:

1. Go to BeagleBoard.org distro page.
2. On distros page, from dropdown select PB2 and download the image.

Tip: You can follow the flash-latest-image guide for more information on flashing the downloaded image to your board.

To see what SW revision is loaded into the eMMC check */etc/dogtag*. It should look something like as shown in example below,

```
root@BeagleBone:~# cat /etc/dogtag
BeagleBoard.org Debian Image 2024-02-24
```

6.5 RMA Support

If you feel your board is defective or has issues, request an Return Merchandise Application (RMA) by filling out the form at http://beagleboard.org/support/rma . You will need the serial number and revision of the board. The serial numbers and revisions keep moving. Different boards can have different locations depending on when they were made. The following figures show the three locations of the serial and revision number.

6.6 Getting Help

If you need some up to date troubleshooting techniques, you can post your queries on link: BeagleBoard.org forum

6.7 Mechanical Details

6.7.1 Dimensions and Weight

Table 6.1: Dimensions & weight

Parameter	Value
Size	56 x 35mm
Max heigh	13.6
PCB Size	55 x 35mm
PCB Layers	10–layers
PCB Thickness	1.6mm
RoHS compliant	Yes
Net Weight	12.7g
Gross Weight	19g