



MPLAB Harmony Help

MPLAB Harmony Integrated Software Framework

Volume II: Supported Hardware

This volume provides information on hardware supported by MPLAB Harmony, including information on devices, development boards, and board support packages.

Description



MPLAB Harmony Board Support Packages provide software support for PIC32 device families and related demonstration development hardware.

A Board Support Package (BSP) provides support for board-specific hardware, which is provided in two forms:

- **BSP Library:** Provides board-specific initialization code and an interface to switches, LEDs, and GPIO pins
- **Initial Configuration:** Provides a starting point for configuration settings for clocks, ports, and selected libraries when creating a new project (or project configuration)

Supported Device Families

This section contains information about how MPLAB Harmony supports each PIC32 device family.

Description

MPLAB Harmony supports all PIC32 device families, with the exception of the PIC32MM family. Middleware, driver, and system service library interfaces are abstracted so that differences between devices are hidden from applications. Library configuration and initialization may contain device-specific options, but these differences are managed by the MPLAB Harmony Configurator (MHC) tool and are not exposed to applications by the library's client interface. However, some libraries may contain optional features that require specific hardware support and are not available on devices that do not provide the required features. Refer to the help section for each library for details on any optional features provided.

Demonstration applications are available for most device families, depending on the availability of development boards that use the device. Applications often have multiple configurations, each of which may support a different development board and device family.

Refer to the Microchip 32-bit design center (<http://www.microchip.com/design-centers/32-bit>) for information on PIC32 devices.

Supported Development Boards

Provides details on the development tools that are used by the Board Support Packages and demonstration applications.

Description

This section provides information on the development tools (i.e., hardware) used by the MPLAB Harmony Board Support Packages and demonstration applications.

ATWINC1500-XSTK Xplained Pro Starter Kit

Provides information on the ATWINC1500-XSTK Xplained Pro Starter Kit.

Description

Microchip Part Number:

ATWINC1500-XSTK

Product Page:

<http://www.microchip.com/developmenttools/productdetails.aspx?partno=atwinc1500-xstk>

Hardware Layout



BM64 Bluetooth Module Daughter Board

Provides information on the BM64 Bluetooth Module Daughter Board.

Description

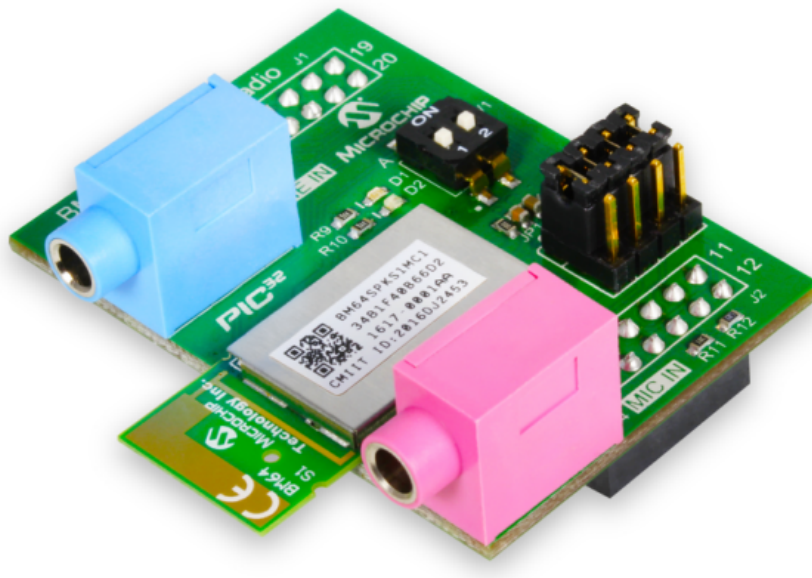
Microchip Part Number:

AC320032-3

Product Page:

<http://www.microchip.com/developmenttools/productdetails.aspx?partno=ac320032-3>

Hardware Layout



CAN/LIN PICtail Plus Daughter Board

Provides information on the CAN/LIN PICtail Plus Daughter Board.

Description

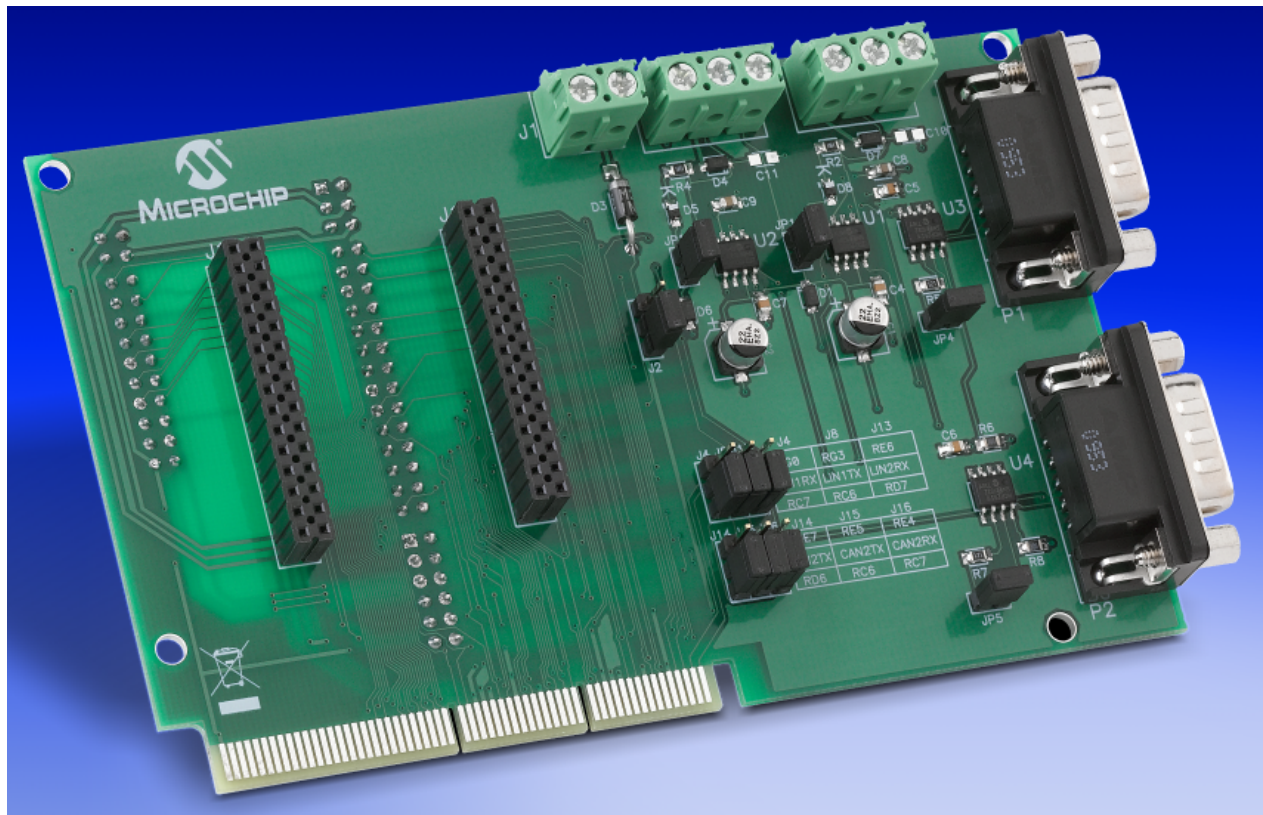
Microchip Part Number:

AC164130-2

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ac164130-2>

Hardware Layout



chipKIT WF32 Wi-Fi Development Board

Provides information on the chipKIT™ WF32™ Wi-Fi Development Board.

Description

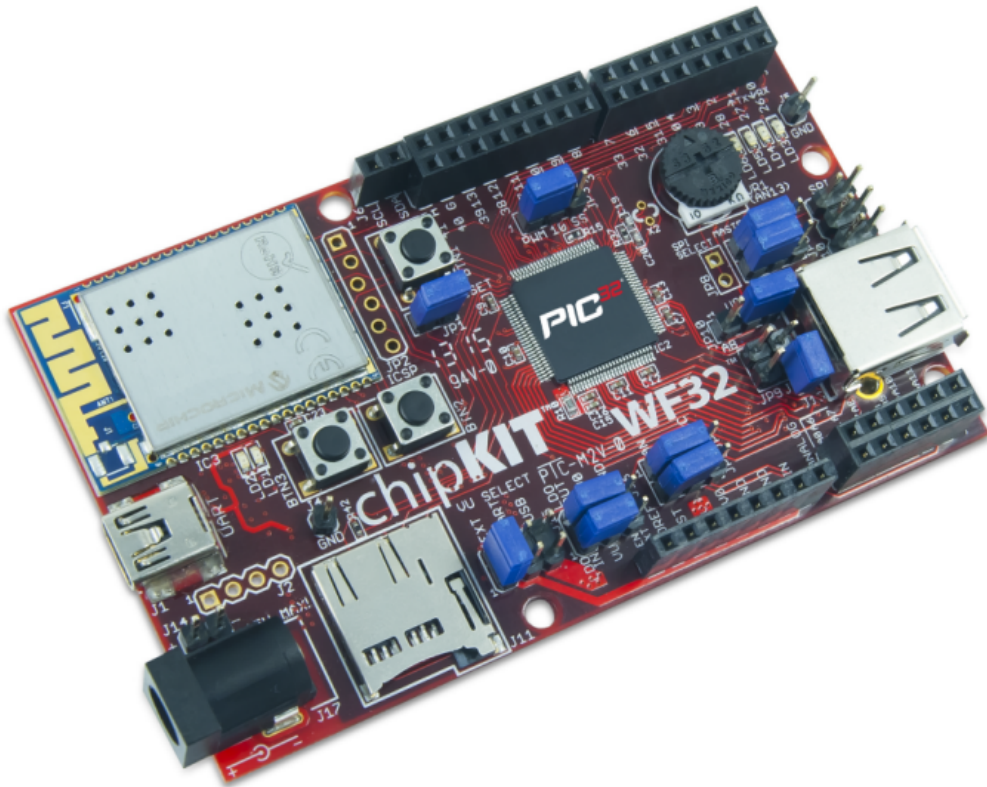
Microchip Part Number:

TDGL021

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=TDGL021>

Hardware Layout



chipKIT Wi-FIRE Development Board

Provides information on the chipKIT™ Wi-FIRE Development Board.

Description

Microchip Part Number:

TDGL021-2

Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=TDGL021-2>

Hardware Layout



dsPICDEM MCLV-2 Development Board

Provides information on the dsPICDEM™ MCLV-2 Development Board.

Description

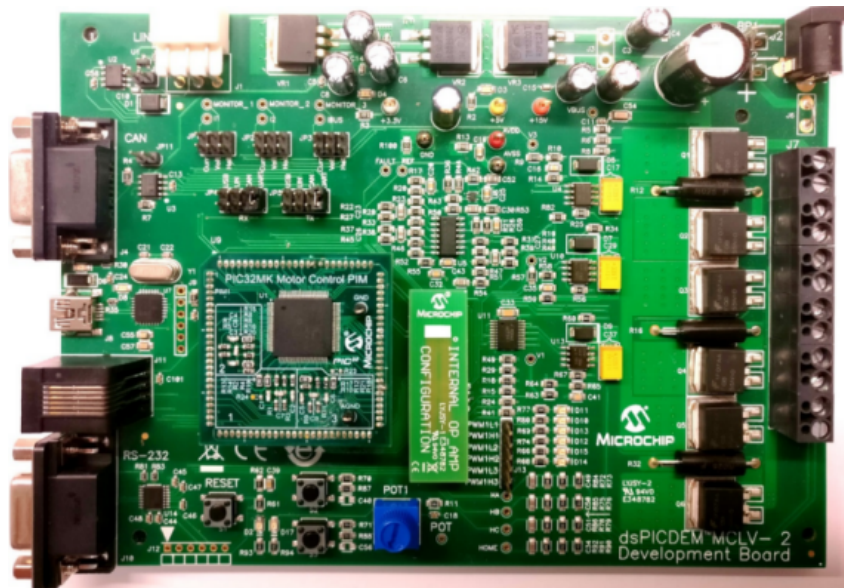
Microchip Part Number:

DM330021-2

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dm330021-2>

Hardware Layout



dsPICDEM MCHV-3 Development Board

Provides information on the dsPICDEM™ MCHV-3 Development Board.

Description

Microchip Part Number:

DM330023-3

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dm330023-3>

Hardware Layout**Explorer 16 Development Board**

Provides information on the Explorer 16 Development Board.

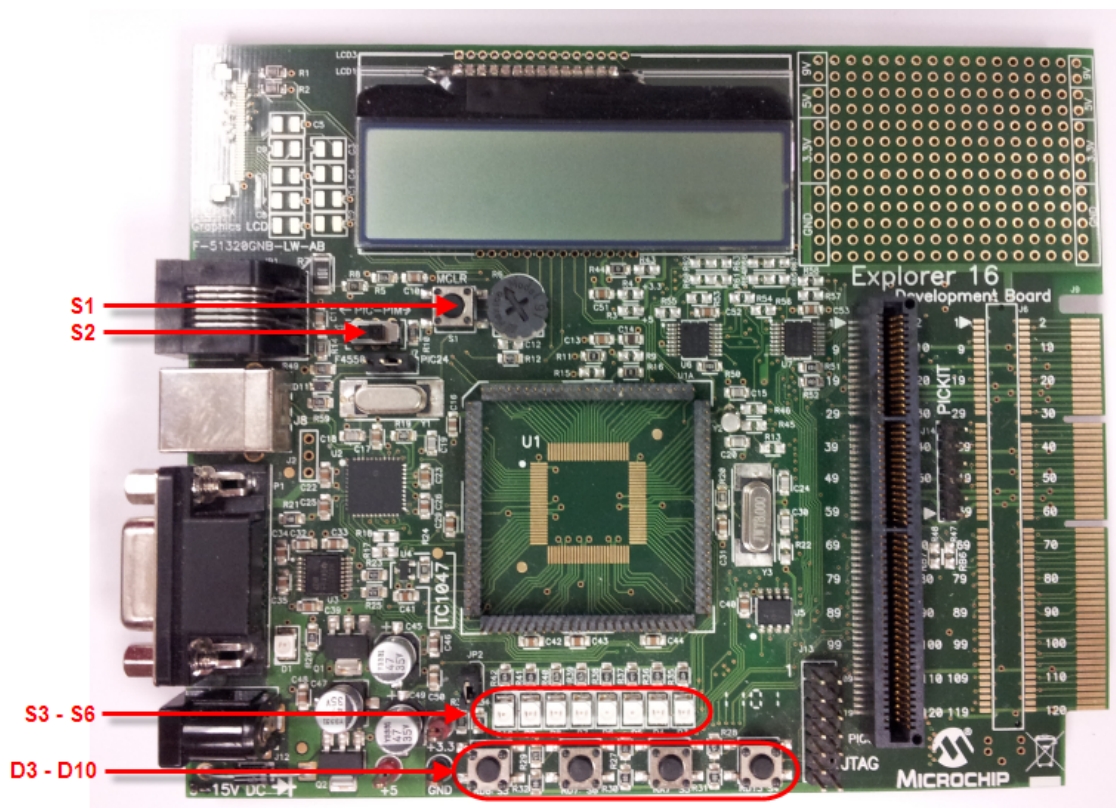
Description**Microchip Part Number:**

DM240001

Product Page:

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en024858&part=DM240001

Hardware Layout



Switch/Button Descriptions

1. S1 - Reset button (MCLR)/
2. S2 - Processor switch. This switch determines which processor is running, the processor on the board or the processor on the Plug-In-Module (PIM).
3. S3, S4, S5, and S6 - Application switches. For information about which pin is connected to this switch, please refer to the Information Sheet for the PIM in use.
4. D3 through D10 - Application LEDs. For information about which pin is connected to this LED, please refer to the Information Sheet for the PIM in use.

Explorer 16/32 Development Board

Provides information on the Explorer 16/32 Development Board.

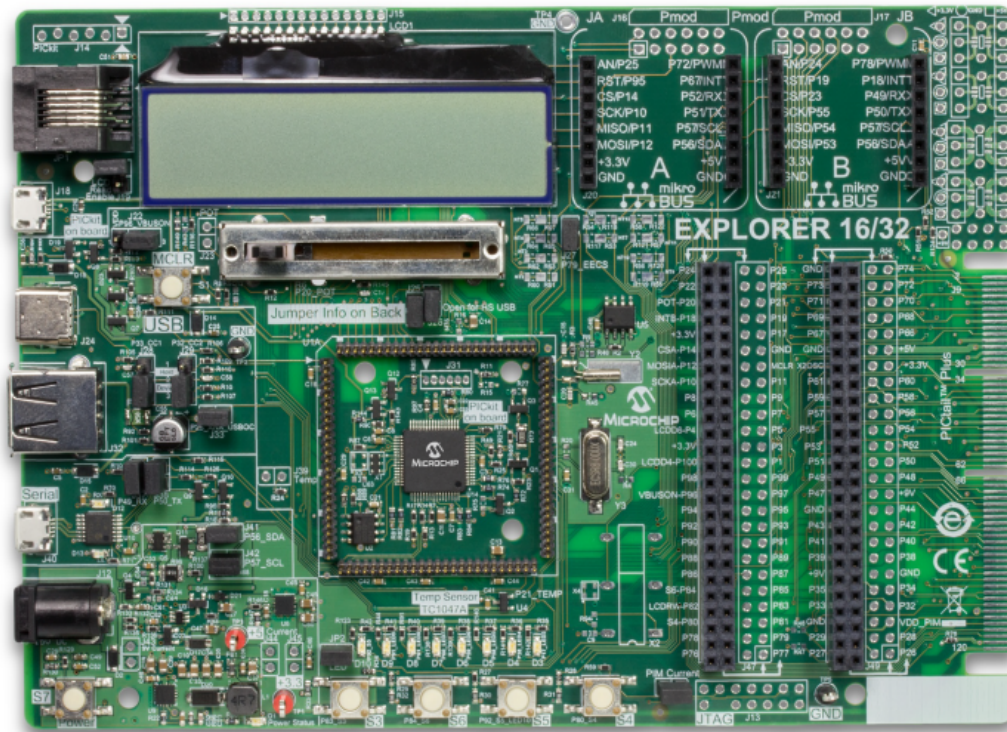
Description

Microchip Part Number:

DM240001-2

Product Page:

<http://www.microchip.com/developmenttools/productdetails.aspx?partno=dm240001-2>



Ethernet PICtail Plus Daughter Board

Provides information on the Ethernet PICtail Plus Daughter Board.

Description

Microchip Part Number:

AC164123

Product Page:

<http://www.microchip.com/AC164123>

Hardware Layout



Fast 100Mbps Ethernet PICtail Plus Daughter Board

Provides information on the Fast 100Mbps Ethernet PICtail Plus Daughter Board.

Description

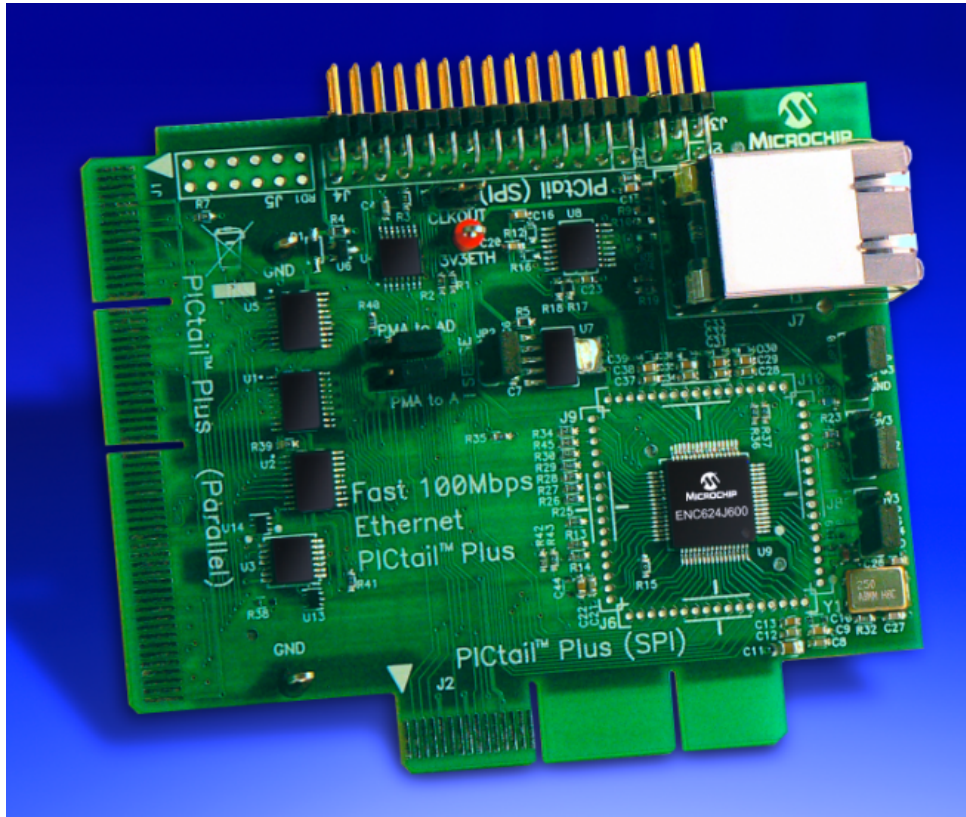
Microchip Part Number:

AC164132

Product Page:

<http://www.microchip.com/AC164132>

Hardware Layout



Graphics Controller PICtail Plus Epson S1D13517 Daughter Board

Provides information about the Graphics LCD Controller PICtail Plus Epson S1D13517 Daughter Board.

Description

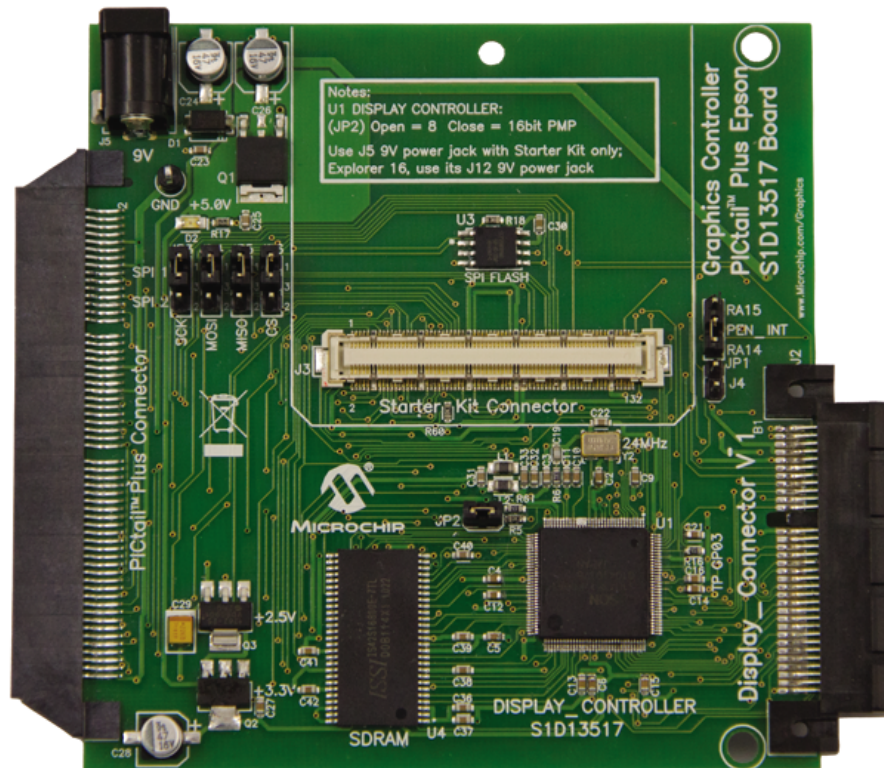
Microchip Part Number:

AC164127-7

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ac164127-7>

Hardware Layout



Graphics Display Truly 3.2" 320x240 Board

Provides information on the Graphics Display Truly 3.2" 320x240 Board.

Description

The Graphics Display Truly 3.2 240x320 Board is a demonstration board for evaluating Microchip graphic display solutions and the Graphics Library for 16- and 32-bit microcontrollers.

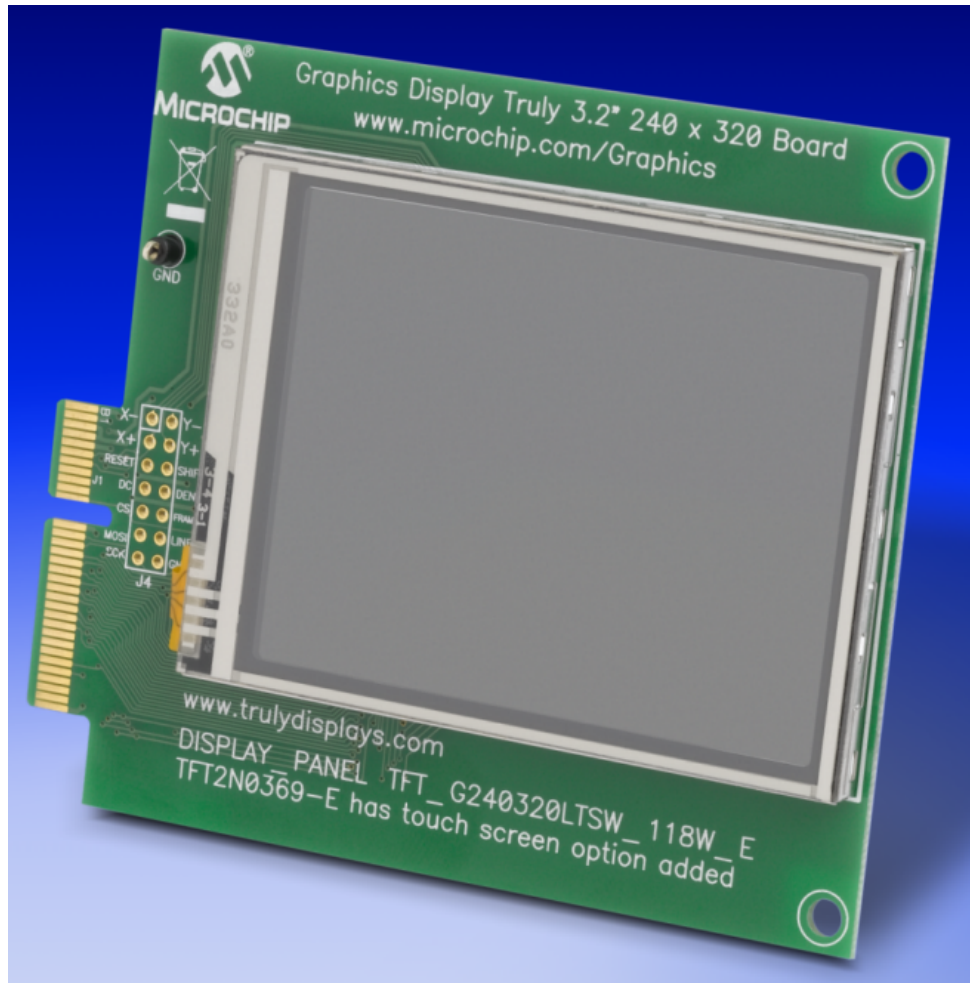
Microchip Part Number

AC164127-4

Product Page

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC164127-4>

Hardware Layout



Graphics Display Powertip 4.3" 480x272 Board

Provides information on the Graphics Display Powertip 4.3" 480x272 Board.

Description

The Graphics Display Powertip 4.3" 480x272 Board is a demonstration board for evaluating Microchip's graphic display solutions and the Graphics Library for 16- and 32- bit microcontrollers.

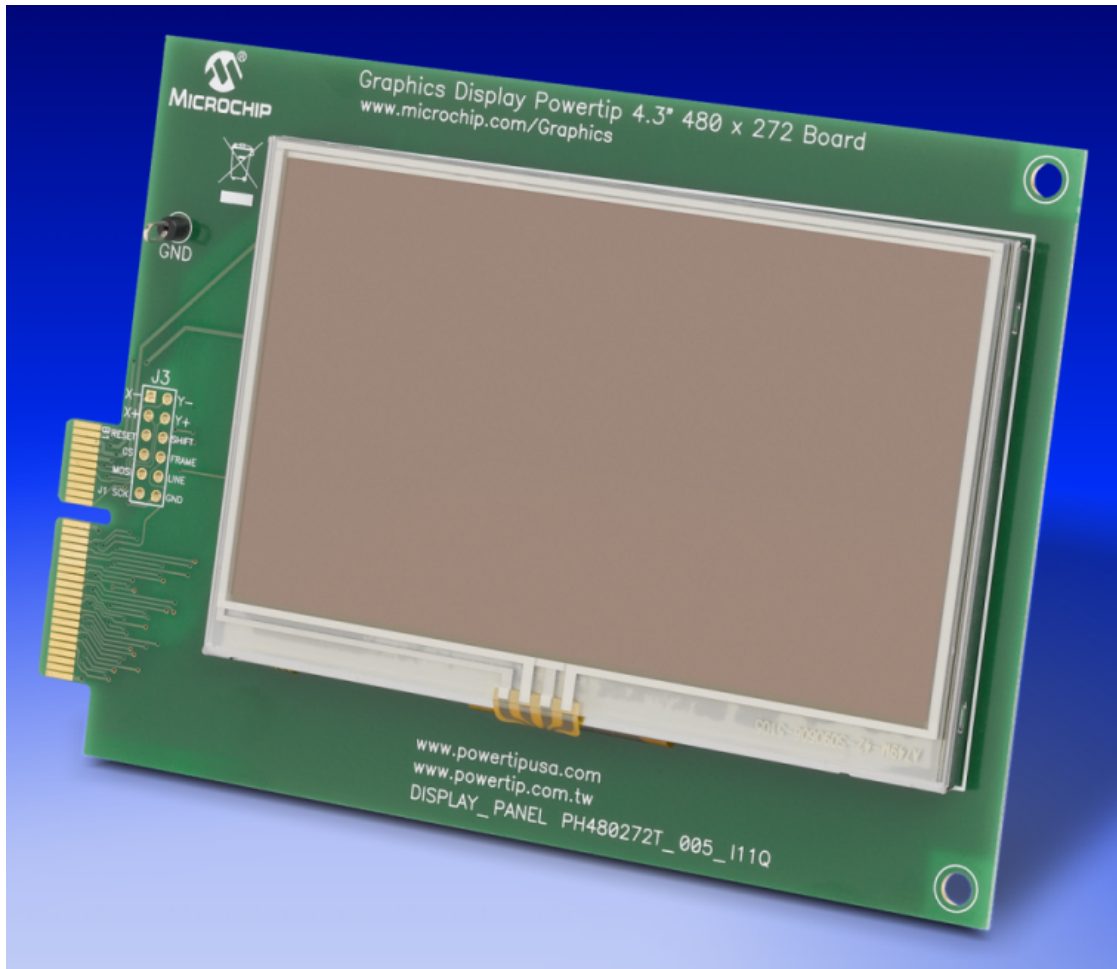
Microchip Part Number

AC164127-6

Product Page

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC164127-6>

Hardware Layout



Graphics Display Truly 5.7" 640x480 Board

Provides information on the Graphics Display Truly 5.7" 640x480 Board.

Description

The Graphics Display Truly 5.7" 640x480 Board is a demonstration board for evaluating Microchip's graphic display solution and Graphics Library for 16- and 32-bit microcontrollers

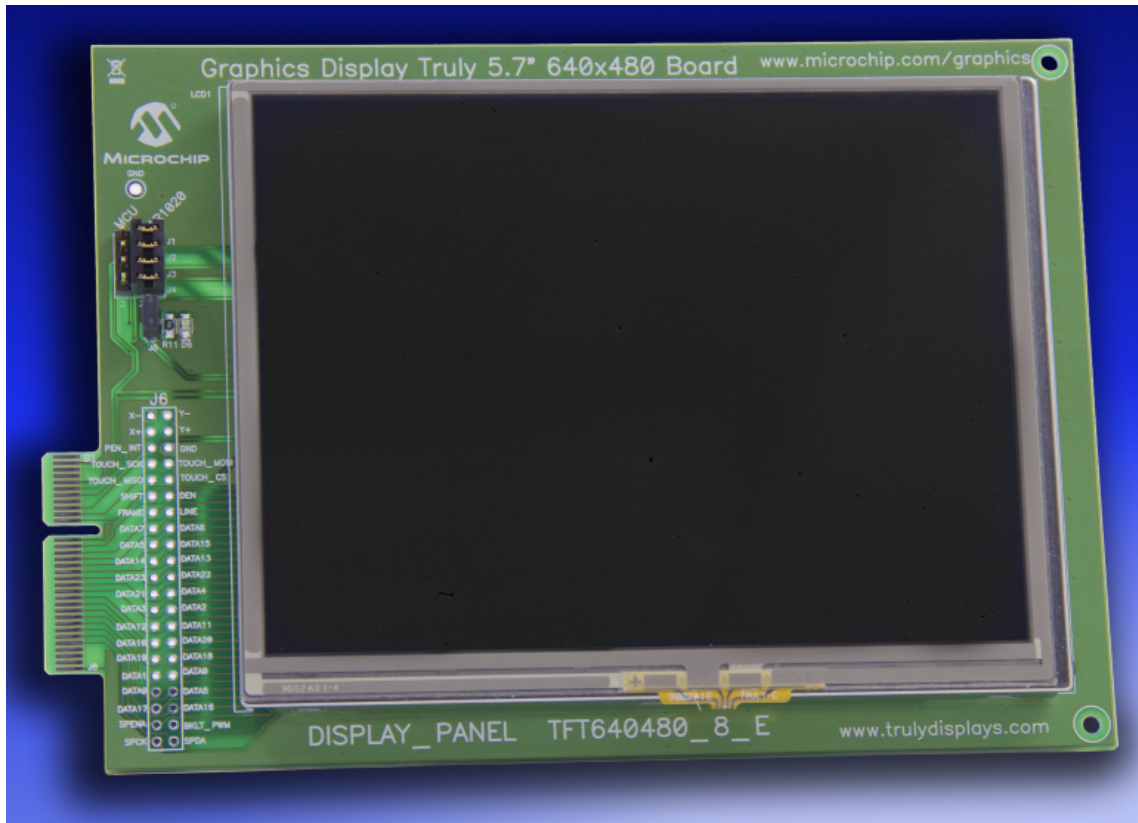
Microchip Part Number

AC164127-8

Product Page

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC164127-8>

Hardware Layout



Graphics Display Truly 7" 800x480 Board

Provides information on the Graphics Display Truly 7" 800x480 Board.

Description

The Graphics Display Truly 7" 800x480 Board is a demonstration board for evaluating Microchip's graphic display solution and Graphics Library for 16- and 32-bit microcontrollers.

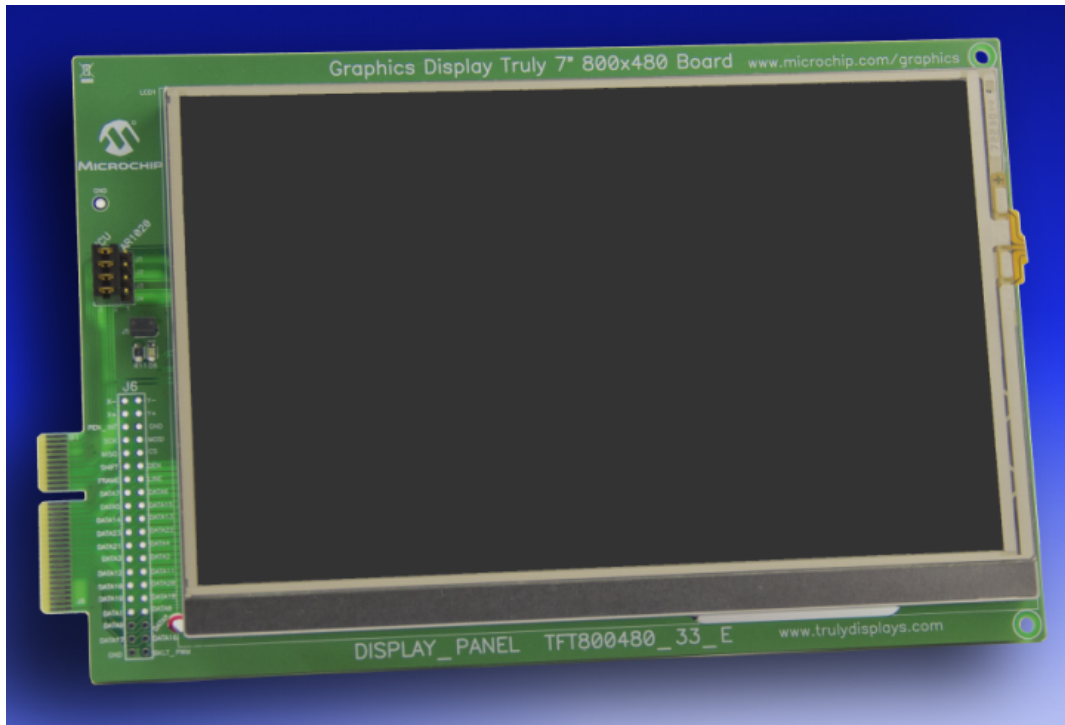
Microchip Part Number

AC164127-9

Product Page

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ac164127-9>

Hardware Layout



Graphics LCD Controller PICtail Plus SSD1926 Daughter Board

Provides information on the Graphics LCD Controller PICtail Plus SSD1926 Daughter Board.

Description

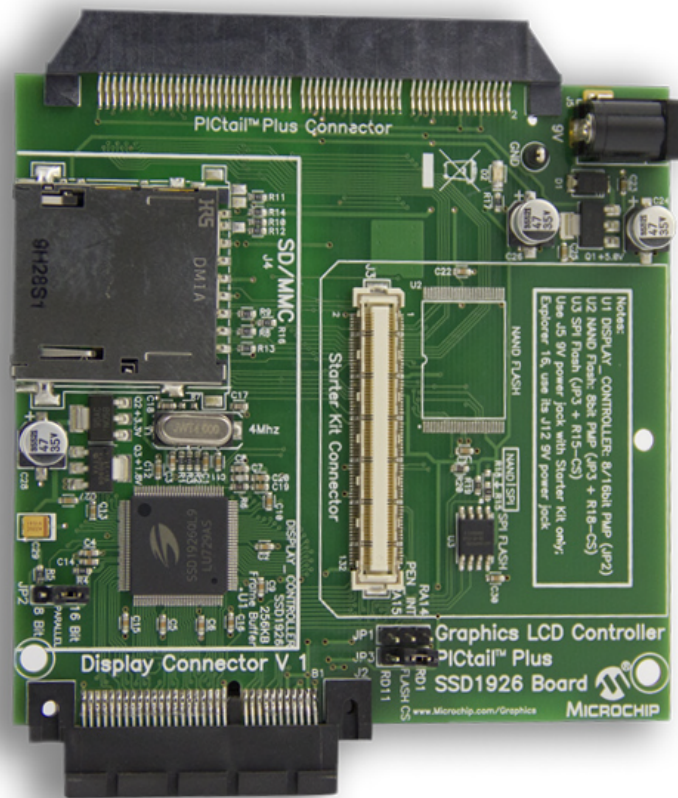
Microchip Part Number:

AC164127-5

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ac164127-5>

Hardware Layout



High-Performance 4.3" WQVGA Display Module with maXTouch

Provides information on the High-Performance 4.3" WQVGA Display Module with maXTouch.

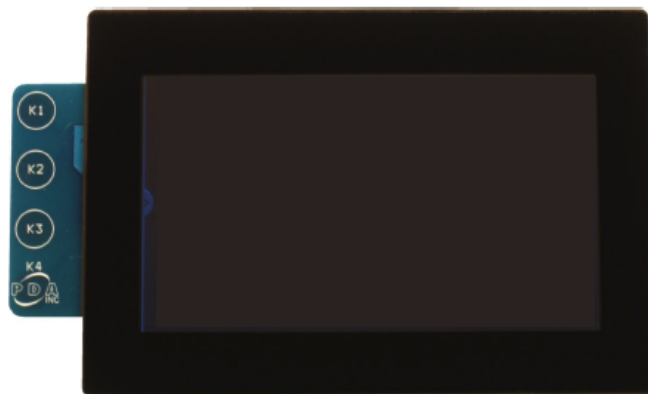
Description

Microchip Part Number:

AC320005-4

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC320005-4>



High-Performance WVGA Display Module with maXTouch

Provides information on the High-Performance 5" WVGA Display Module with maXTouch

Description

Microchip Part Number:

AC320005-5

Microchip Product Page:<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=AC320005-5>

I/O1 Xplained Pro Extension Kit

Provides information on the I/O1 Xplained Pro Extension Kit.

Description

Microchip Part Number:

ATIO1-XPRO

Product Page:<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ATIO1-XPRO>

Low-Cost Controllerless (LCC) Graphics PICtail Plus Daughter Board

Provides information on the Low-Cost Controllerless (LCC) Graphics PICtail Plus Daughter Board.

Description

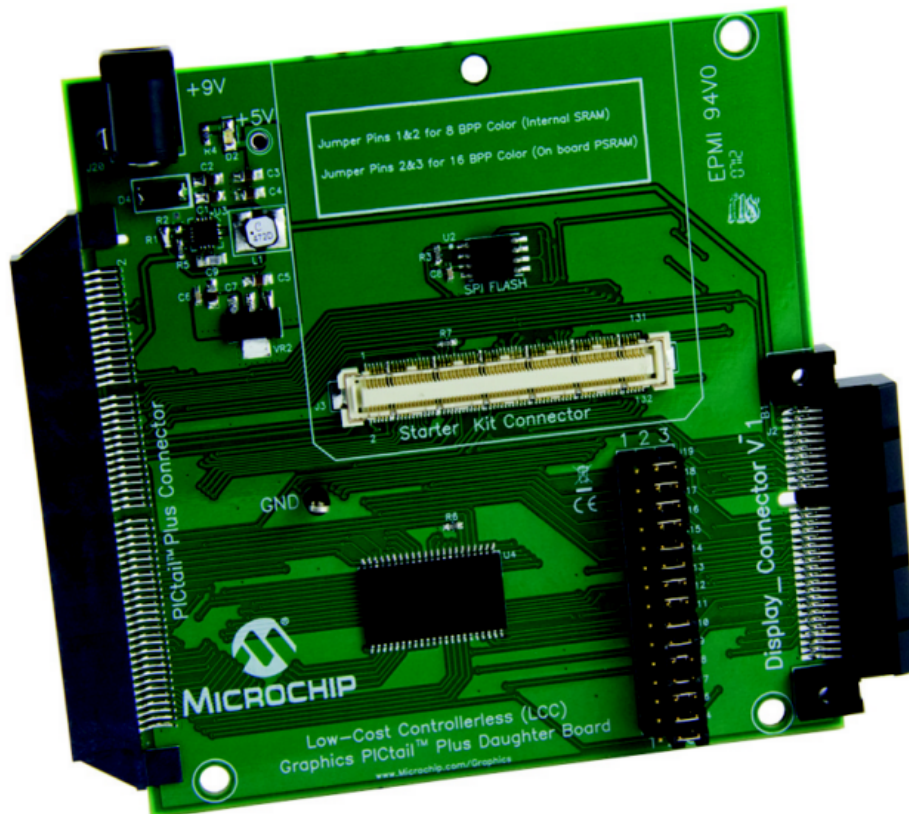
Microchip Part Number:

AC164144

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ac164144>

Hardware Layout



maXTouch Xplained Pro

Provides information on the maXTouch Xplained Pro 3.5" LCD Display with maXTouch sensor.

Description

Microchip Part Number:

ATMXT-XPRO

Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=ATMXT-XPRO>

Hardware Layout



MCP2200 Breakout Module

Provides information on the MCP2200 Breakout Module.

Description

The MCP2200 Breakout Module is a development and evaluation platform for the USB-to-UART serial converter MCP2200 device. The module is comprised of a single Dual In-Line Package (DIP) form-factor board.

Microchip Part Number:

ADM00393

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ADM00393>

Hardware Layout



MikroElektronika WiFi 7 Click Board

Provides information on the MikroElektronika WiFi 7 Click Board with the on-board WINC1500 Wi-Fi module.

Description

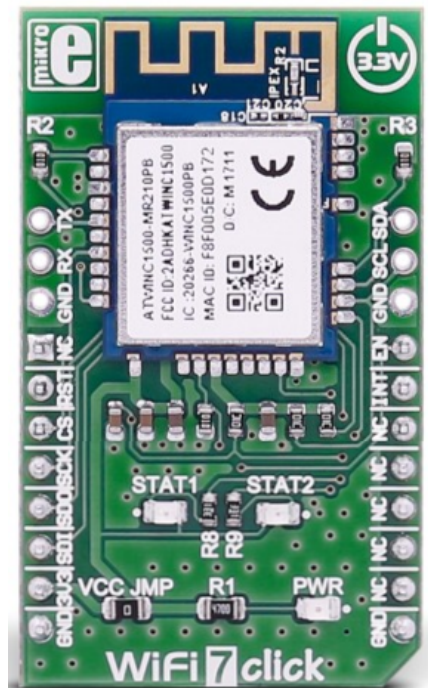
MikroElektronika Part Number:

MIKROE-2046

Product Page:

<https://shop.mikroe.com/click/wireless-connectivity/wifi-7>

Hardware Layout



MPLAB REAL ICE

Provides information on the MPLAB REAL ICE In-Circuit Emulator.

Description

Microchip Part Number:

DV244005

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dv244005>



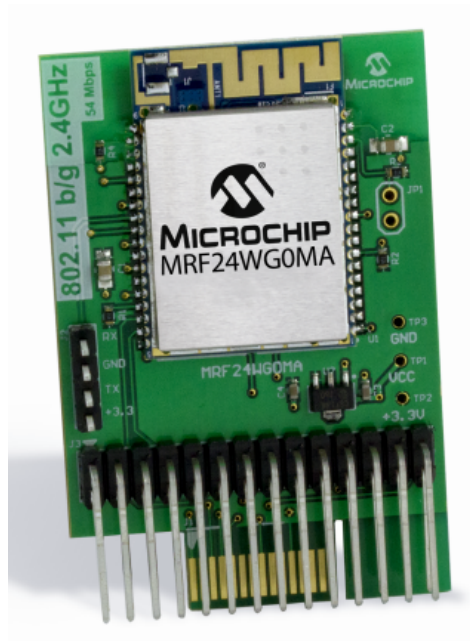
MRF24WG0MA Wi-Fi G PICtail/PICtail Plus Daughter Board

Provides information on the MRF24WG0MA Wi-Fi G PICtail/PICtail Plus Daughter Board.

Description

Microchip Part Number:

AC164149

Microchip Product Page:<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC164149>**Hardware Layout****MRF24WN0MA Wi-Fi PICtail/PICtail Plus Daughter Board**

Provides information on the MRF24WN0MA Wi-Fi PICtail/PICtail Plus Daughter Board.

Description**Microchip Part Number:**

AC164153

Microchip Product Page:<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC164153>**Hardware Layout****Multimedia Expansion Board (MEB)**

Provides information on the Multimedia Expansion Board (MEB).

Description

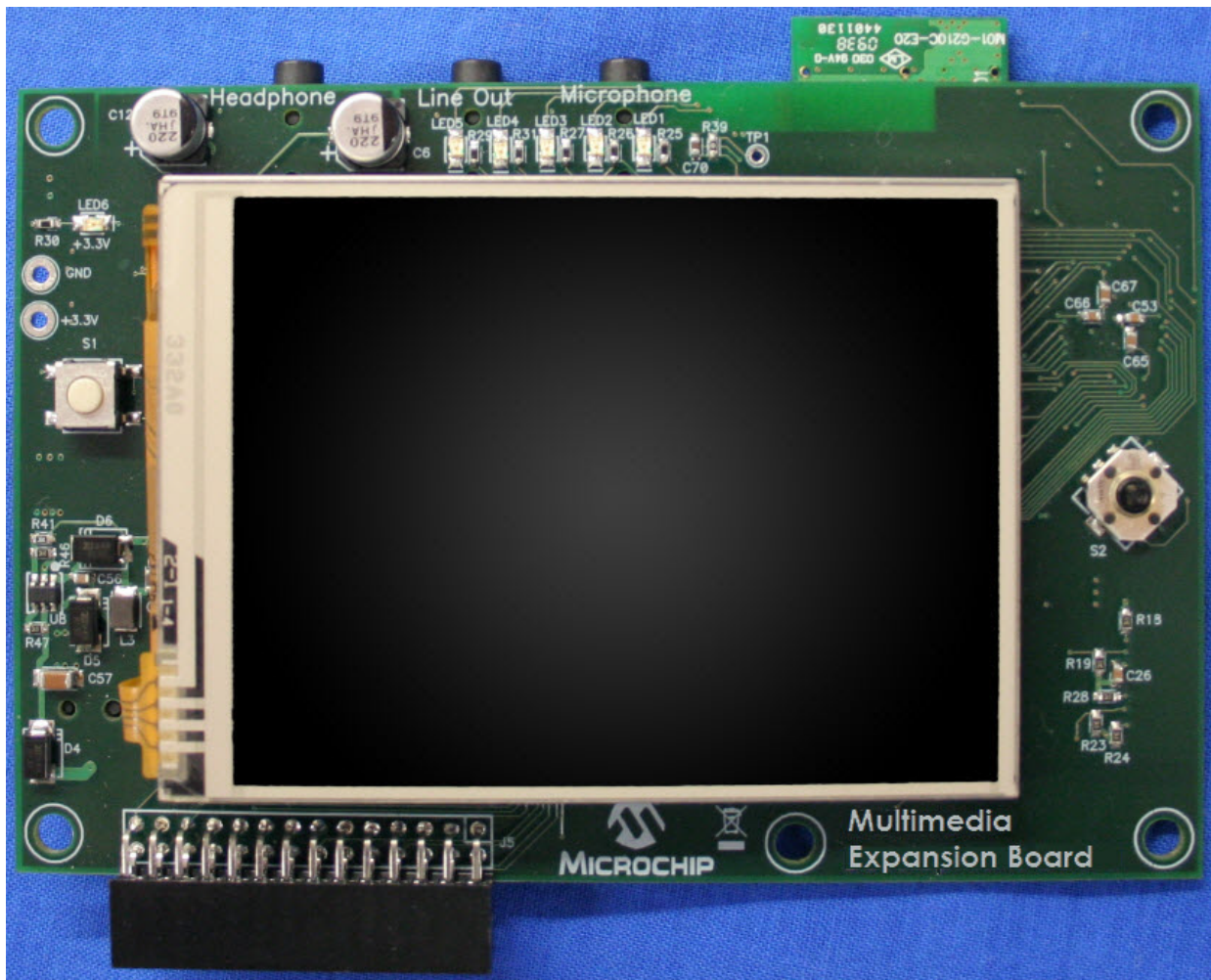
Microchip Part Number:

DM320005

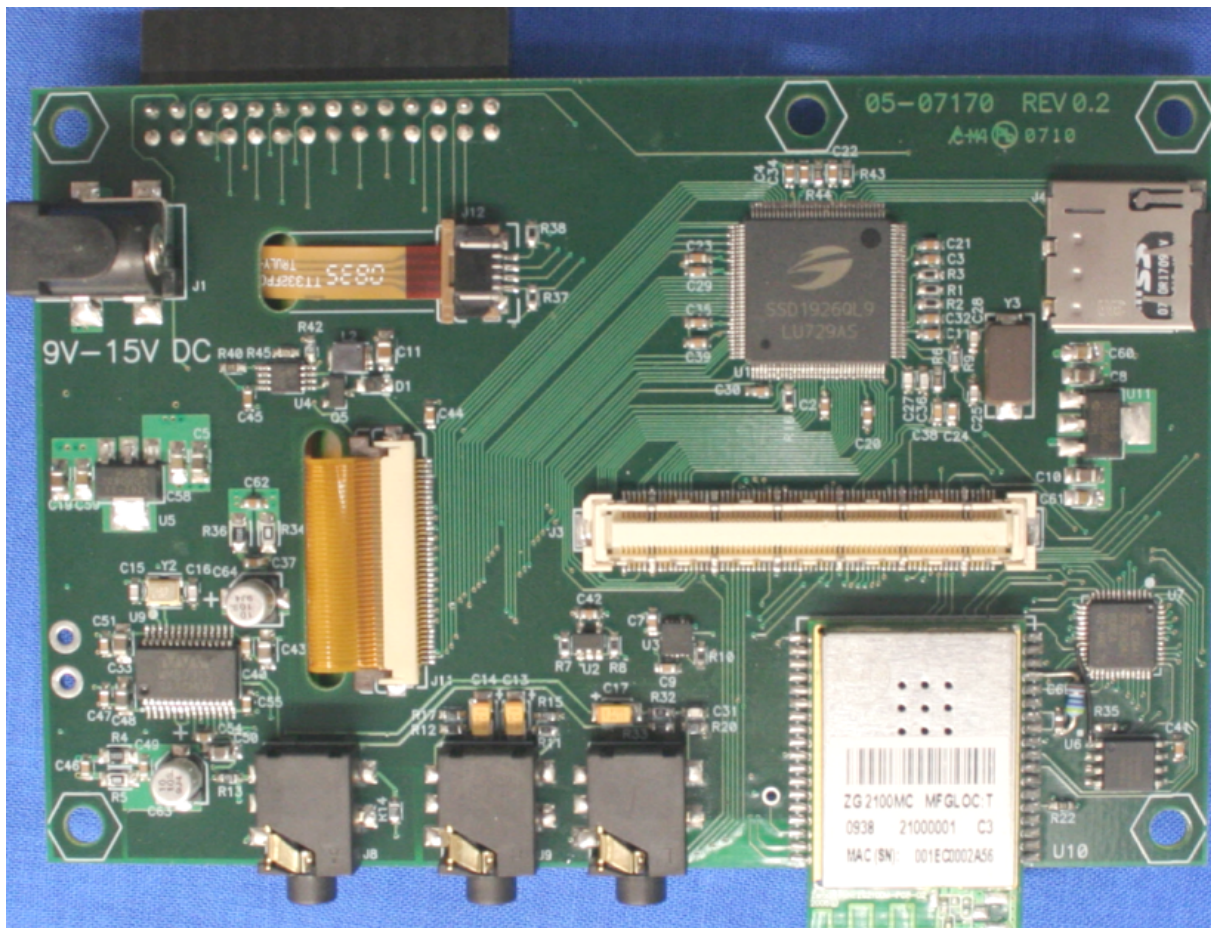
Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320005>

Hardware Layout (Front)



Hardware Layout (Back)



Multimedia Expansion Board II (MEB II)

Provides information on the updated Multimedia Expansion Board II (MEB II).

Description



Note:

Currently, two versions of the MEB II are available for purchase; however, the original version of this product is not recommended for use. Instead, use the updated version of this board, which features a 4.3" WQVGA maXTouch display.

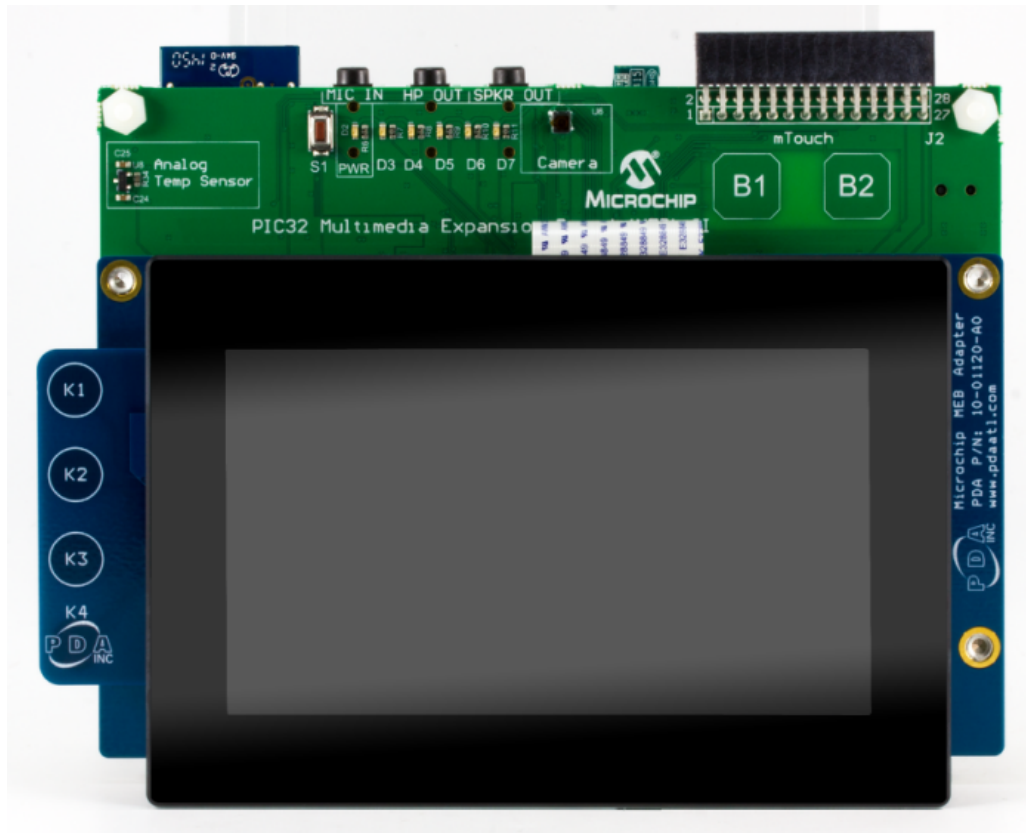
Microchip Part Number

DM320005-5

Product Page

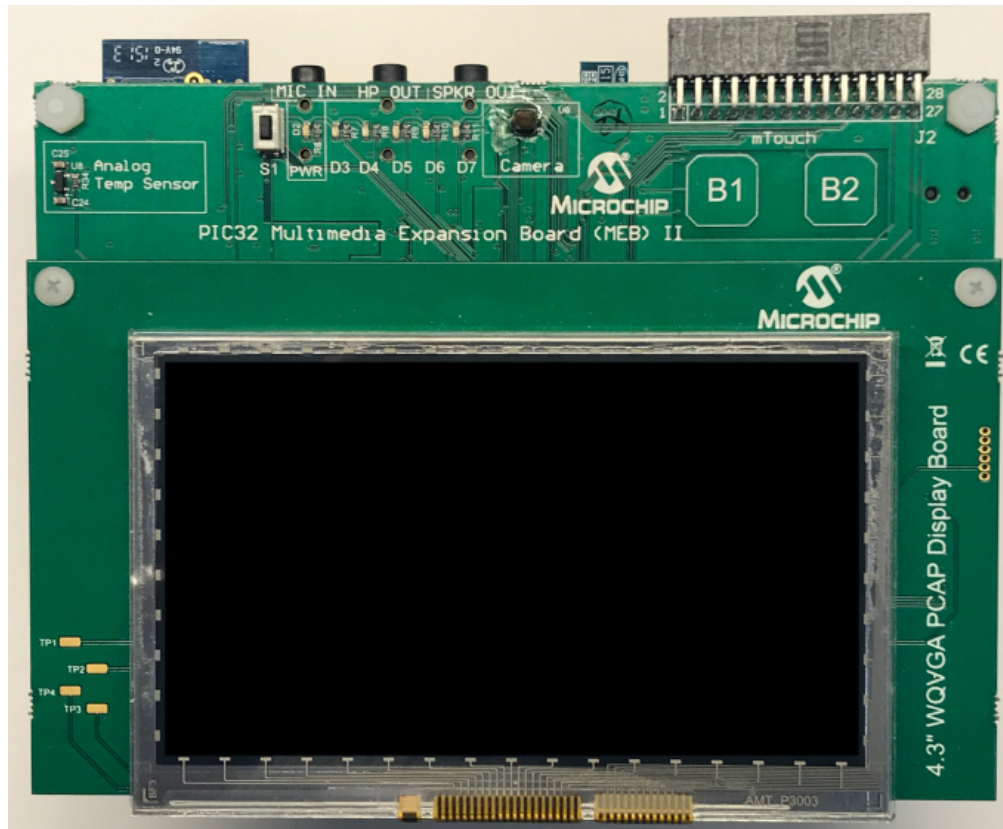
<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dm320005-5>

Hardware Layout

**Microchip Part Number**

DM320005-2

Product Page<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dm320005-2>**Hardware Layout**



OLED1 Xplained Pro Extension Kit

Provides information on the OLED1 Xplained Pro Extension Kit.

Description

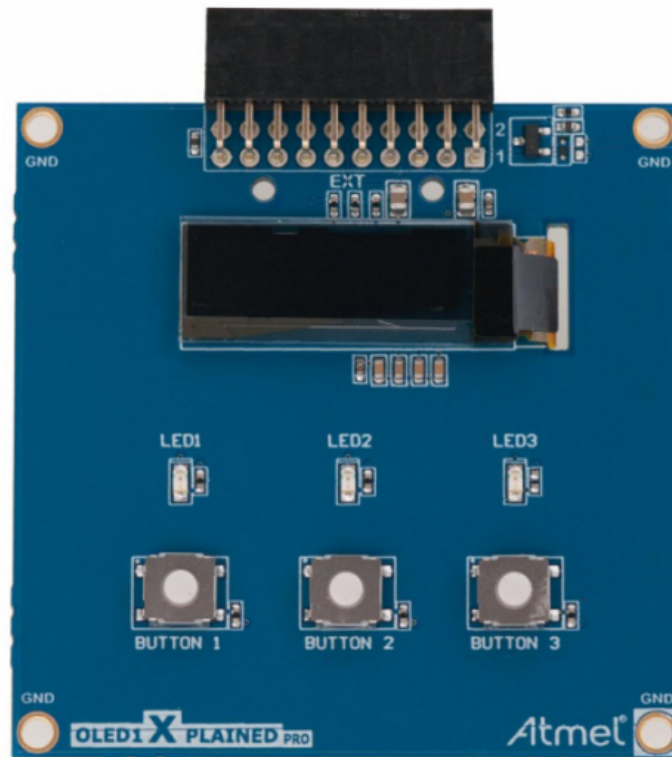
Microchip Part Number:

ATOLED1-XPRO

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=ATOLED1-XPRO>

Hardware Layout



PIC32 Audio DAC Daughter Board - AK4384VT

Provides information on the PIC32 Audio DAC Daughter Board (AK4384VT).

Description

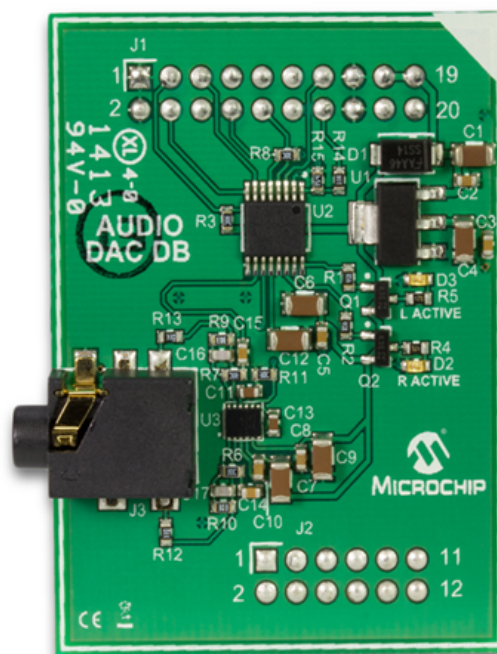
Microchip Part Number:

AC320032-2

Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC320032-2>

Hardware Layout



PIC32 Audio Codec Daughter Board - AK4642EN

Provides information on the PIC32 Audio Codec Daughter Board - AK4642EN.

Description

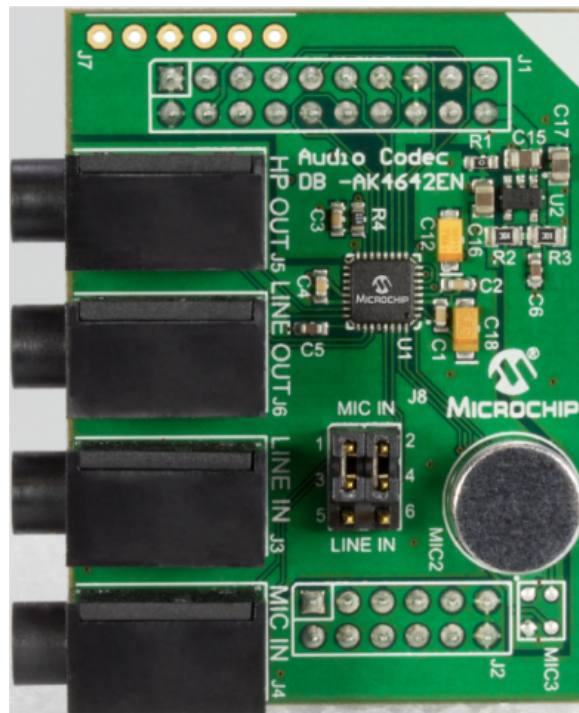
Microchip Part Number:

AC320100

Product Page:

<http://www.microchip.com/AC320100>

Hardware Layout



PIC32 Audio Codec Daughter Board - AK4954A

Provides information on the PIC32 Audio Codec Daughter Board - AK4954A.

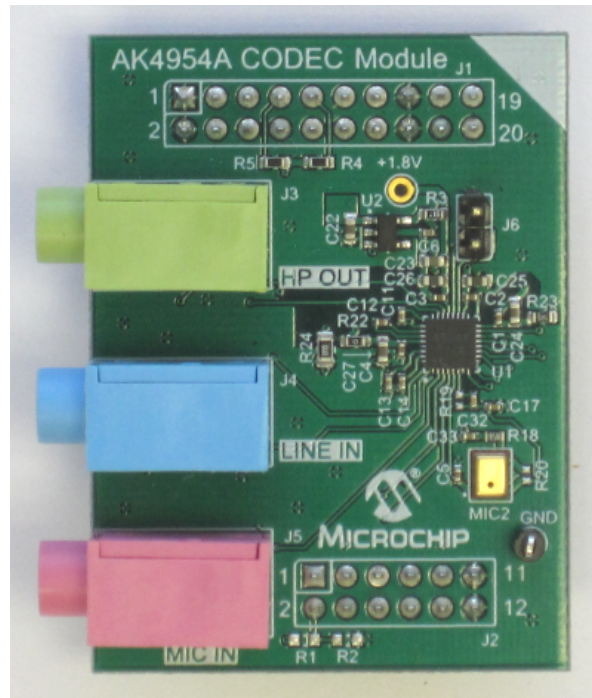
Description



Note:

Please contact your local Microchip sales office for information on obtaining this board.

Hardware Layout



PIC32 Audio Codec Daughter Board - AK7755

Provides information on the PIC32 Audio Codec Daughter Board - AK7755

Description

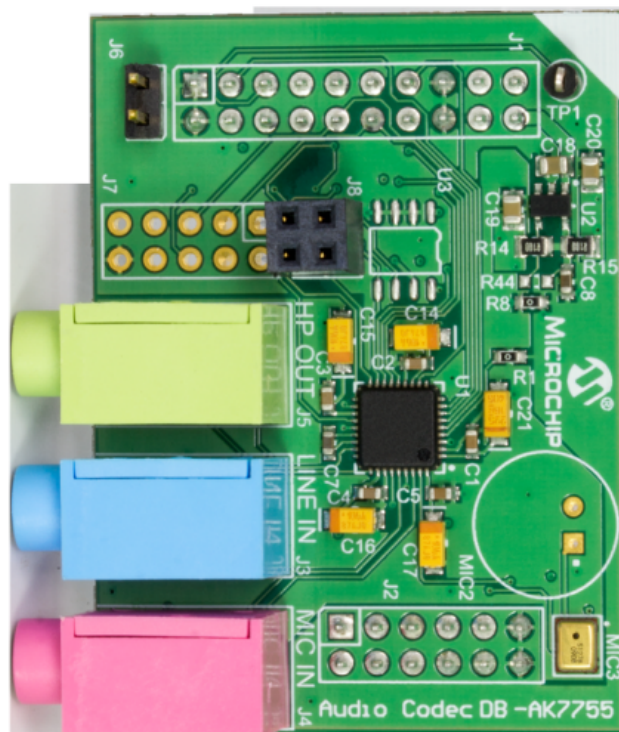
Microchip Part Number:

AC320100

Product Page:

<http://www.microchip.com/AC327755>

Hardware Layout



Provides information on the PIC32 Bluetooth Audio Development Kit.

Description

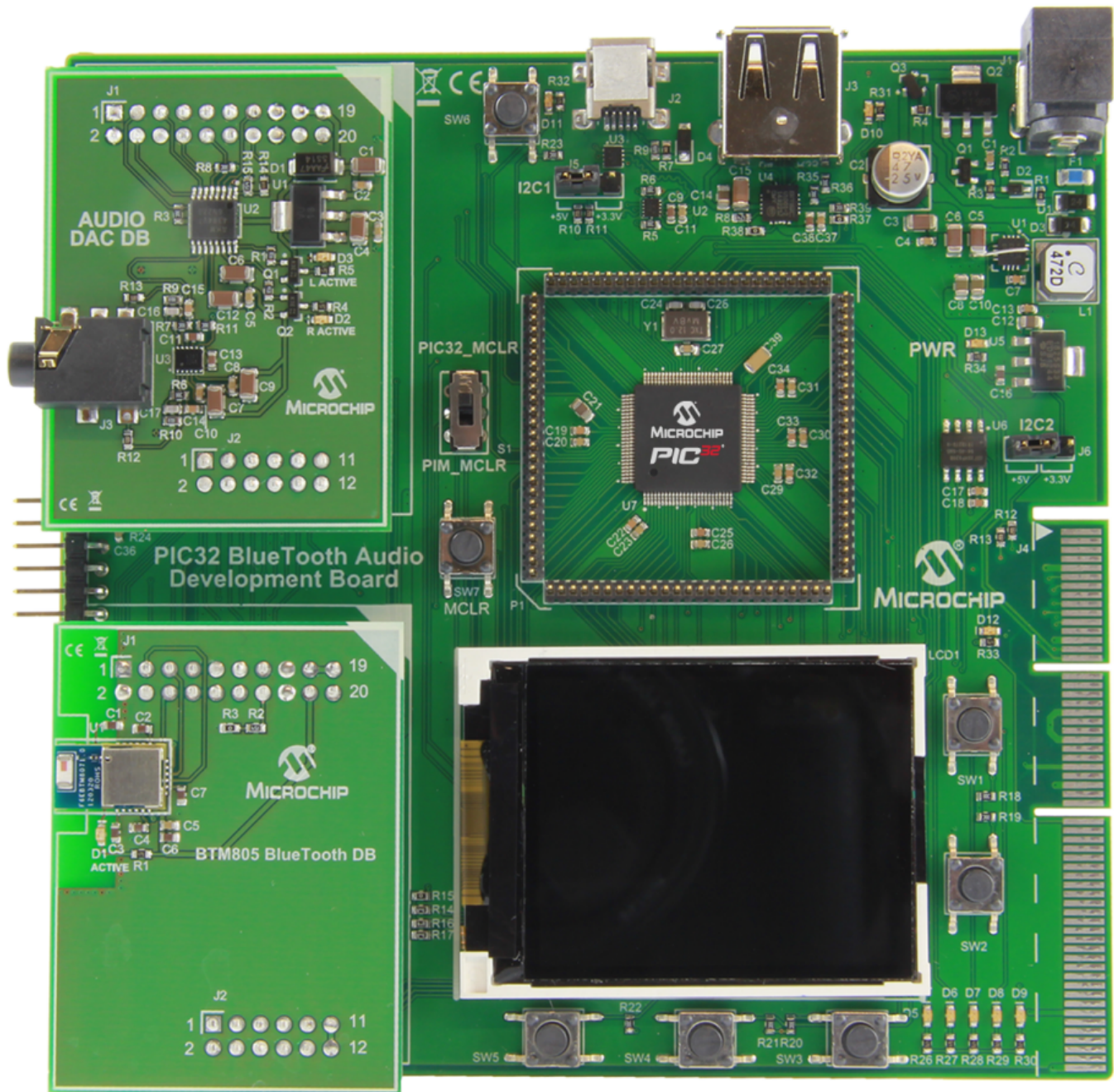
Microchip Part Number:

DV320032

Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DV320032>

Hardware Layout



PIC32 Bluetooth Starter Kit

Provides information on the PIC32 Bluetooth Starter Kit.

Description

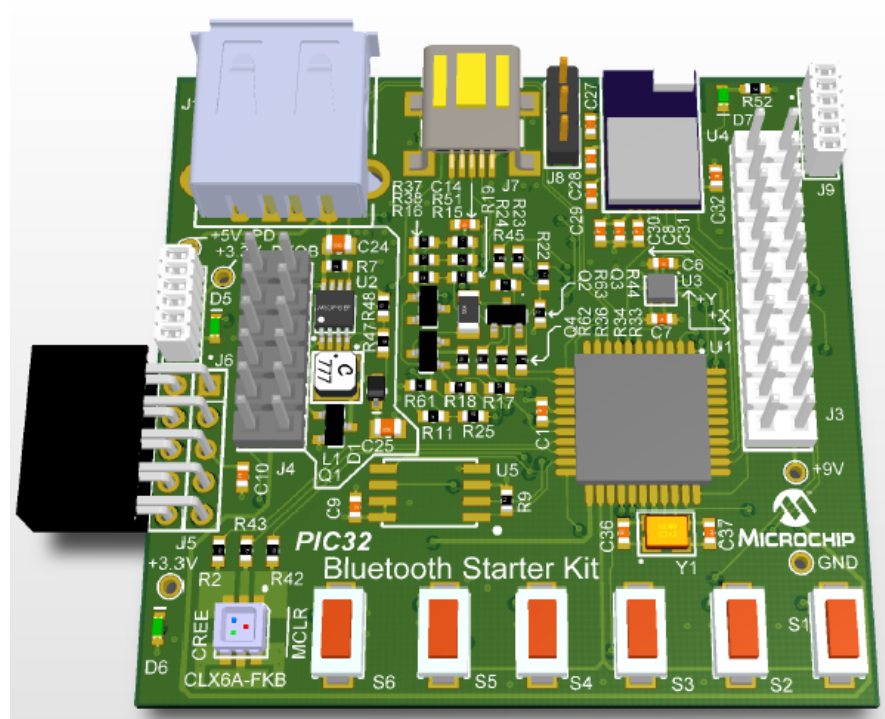
Microchip Part Number:

DM320018

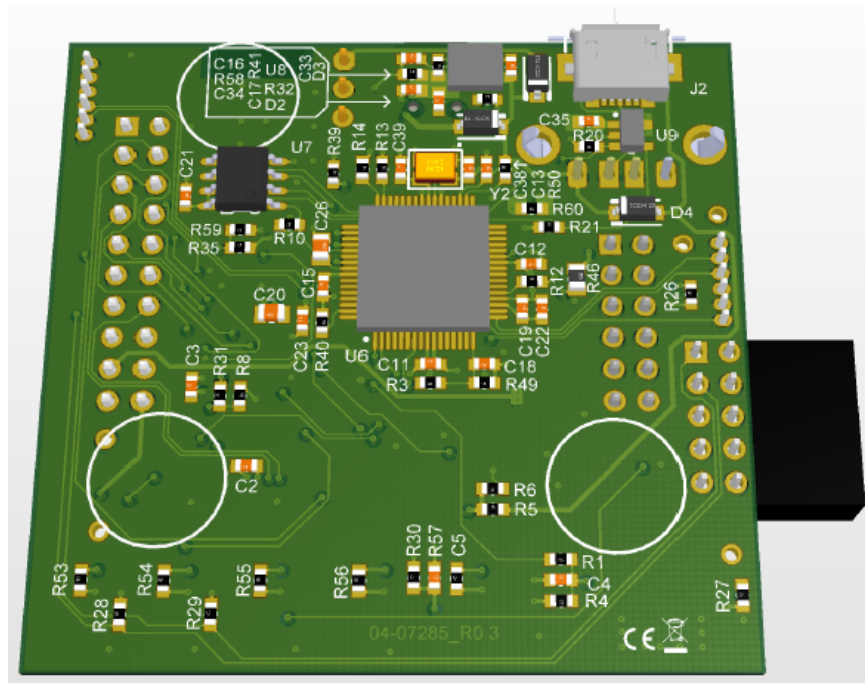
Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320018>

Hardware Layout (Front)



Hardware Layout (Back)



PIC32 Ethernet Starter Kit

Provides information on the PIC32 Ethernet Starter Kit.

Description

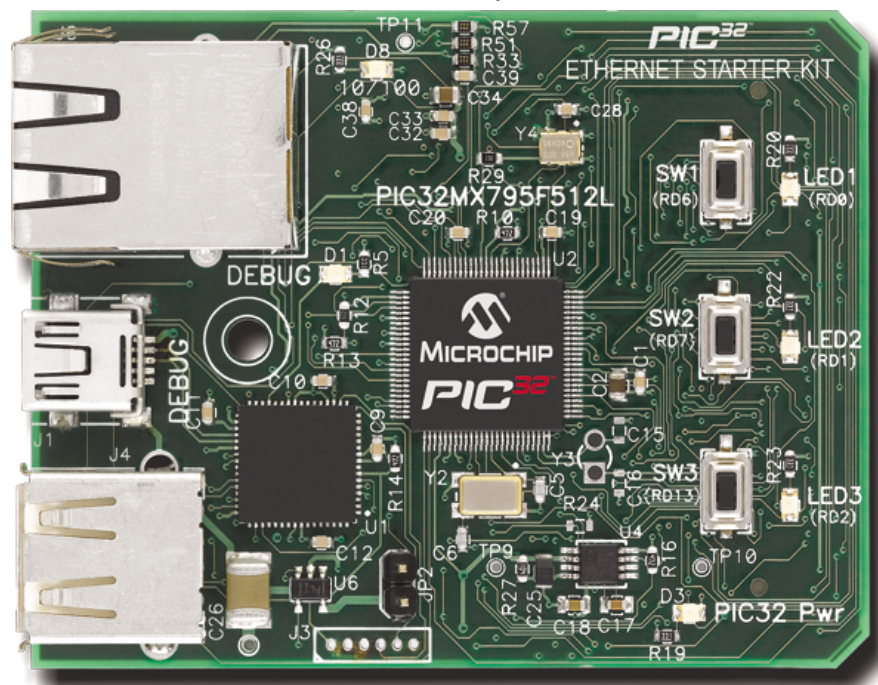
Microchip Part Number:

DM320004

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dm320004>

Hardware Layout



PIC32 Ethernet Starter Kit II

Provides information on the PIC32 Ethernet Starter Kit II.

Description

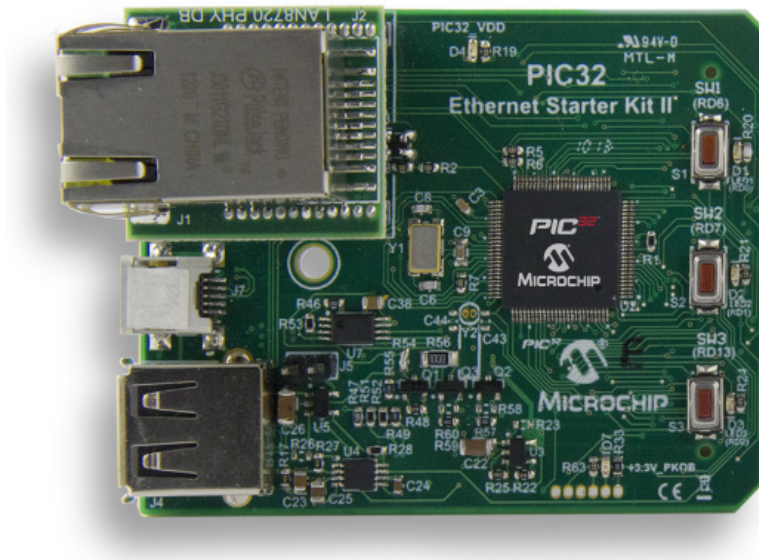
Microchip Part Number:

DM320004-2

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=DM320004-2>

Hardware Layout



PIC32 Graphics Discovery Development Board

Provides information on the PIC32 Graphics Discovery Development Board.

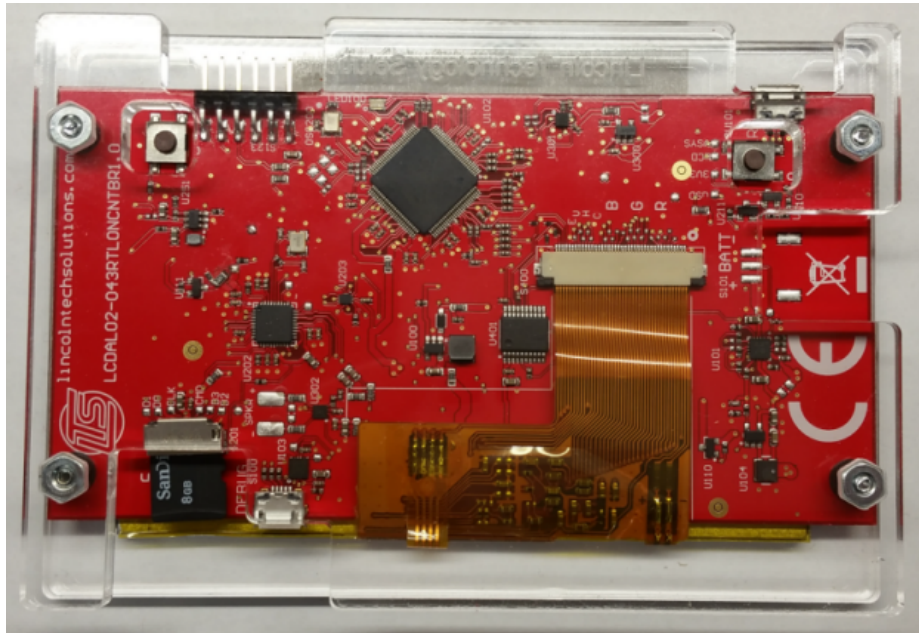
Description



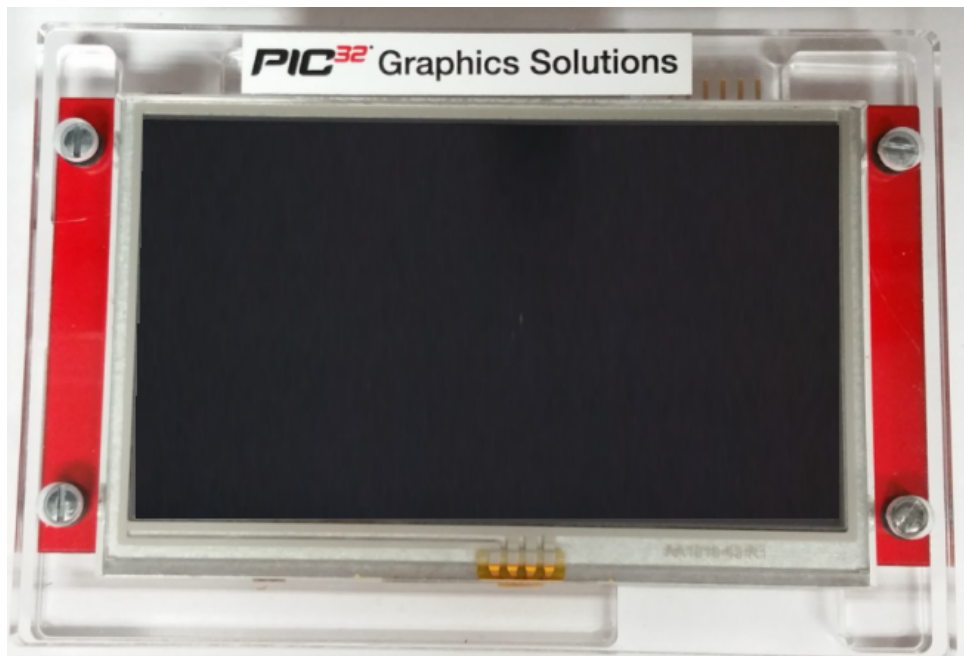
Note:

Please contact your local Microchip sales office for information on obtaining this development board.

Hardware Layout (Front)



Hardware Layout (Back)



PIC32 GUI Development Board with Projected Capacitive Touch

Provides information on the PIC32 GUI Development Board with Projected Capacitive Touch.

Description

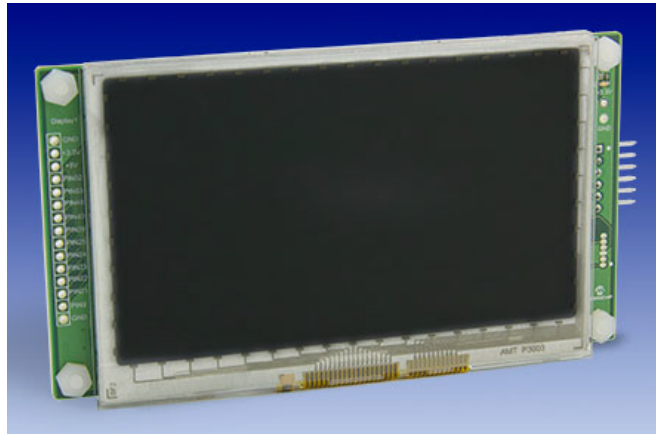
Microchip Part Number:

DM320015

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dm320015>

Hardware Layout



PIC32 USB Digital Audio Accessory Board

Provides information on the PIC32 USB Digital Audio Board.

Description

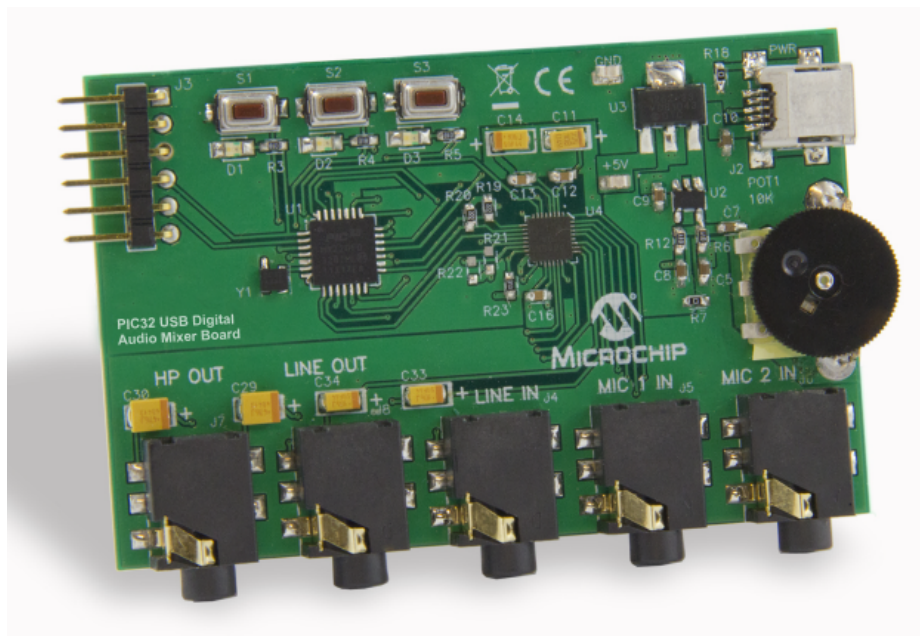
Microchip Part Number:

DM320014

microchipDIRECT Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dm320014>

Hardware Layout



PIC32 USB Starter Kit II

Provides information on the PIC32 USB Starter Kit II.

Description

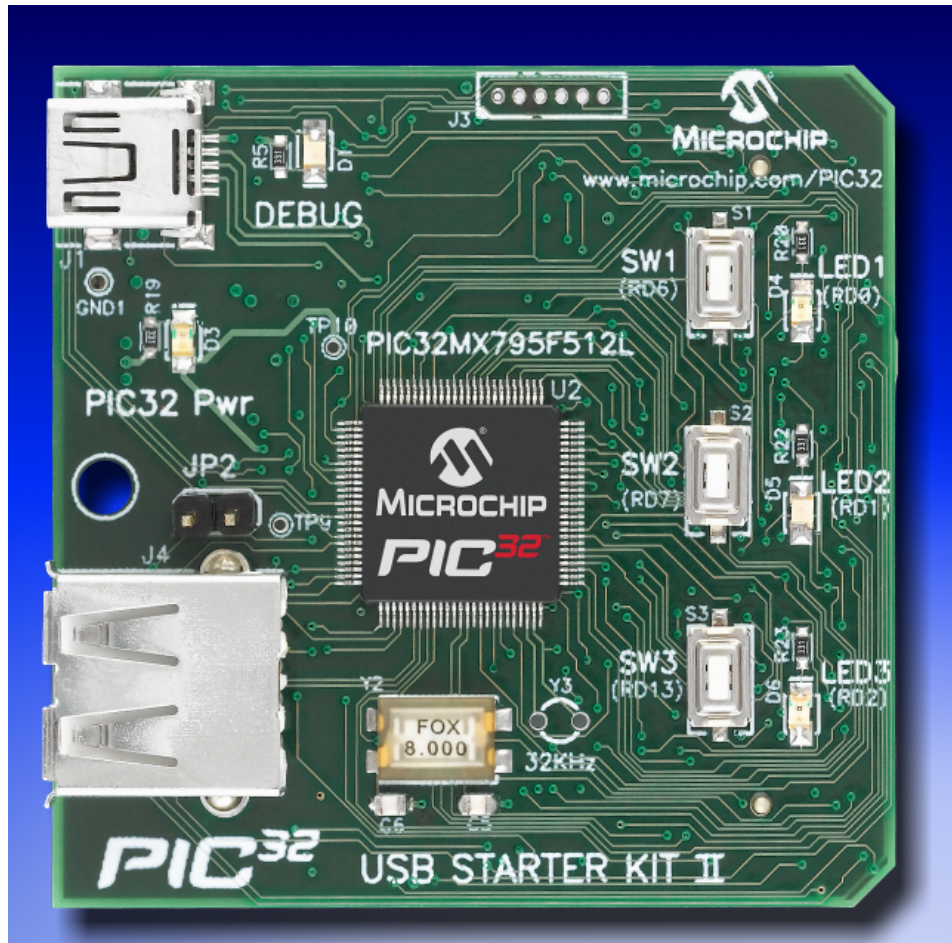
Microchip Part Number:

DM320003-2

Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=dm320003-2>

Hardware Layout



PIC32 USB Starter Kit III

Provides information on the PIC32 USB Starter Kit III.

Description

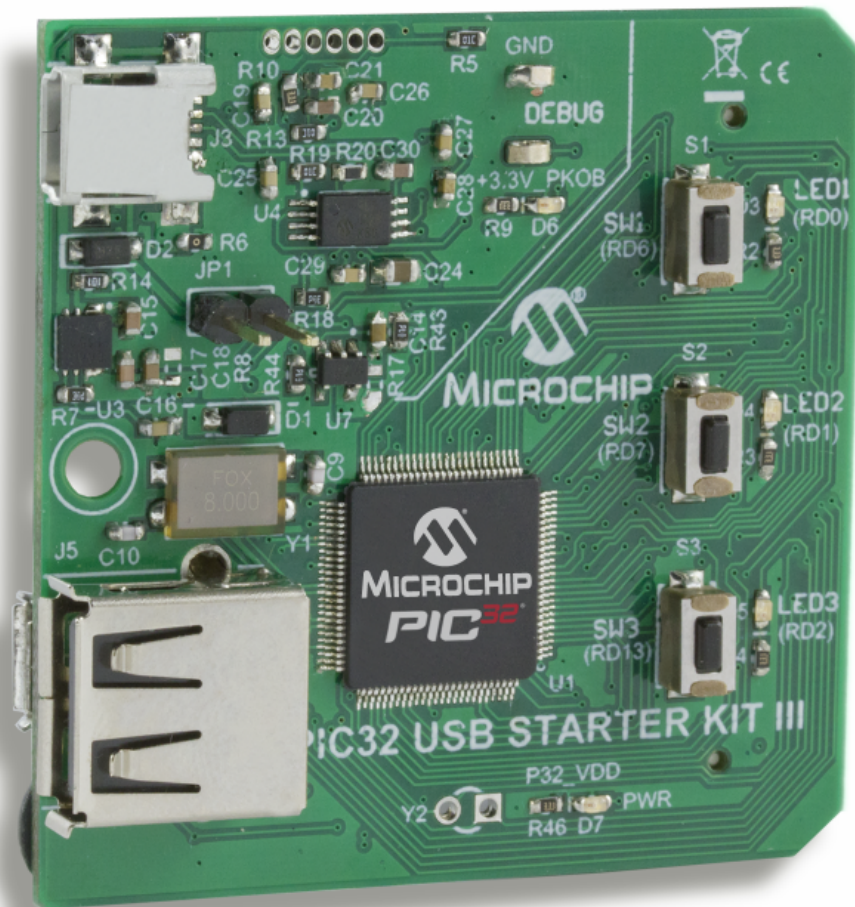
Microchip Part Number:

DM320003-3

Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320003-3>

Hardware Layout



PIC32MK General Purpose (GP) Development Board

Provides information on the PIC32MK General Purpose (GP) Development Board.

Description

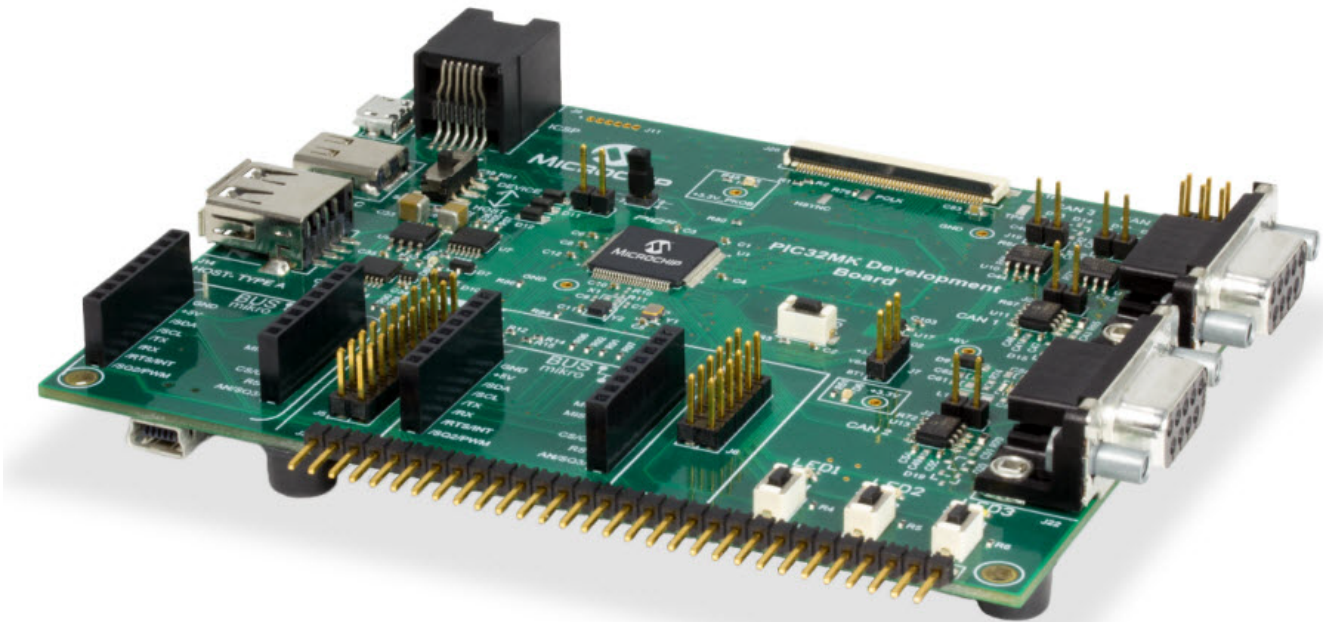
Microchip Part Number:

DM320106

Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320106>

Hardware Layout



PIC32MK 100-pin Motor Control Plug-in Module (PIM)

Provides information on the PIC32MK 100-pin Motor Control Plug-in Module (PIM).

Description

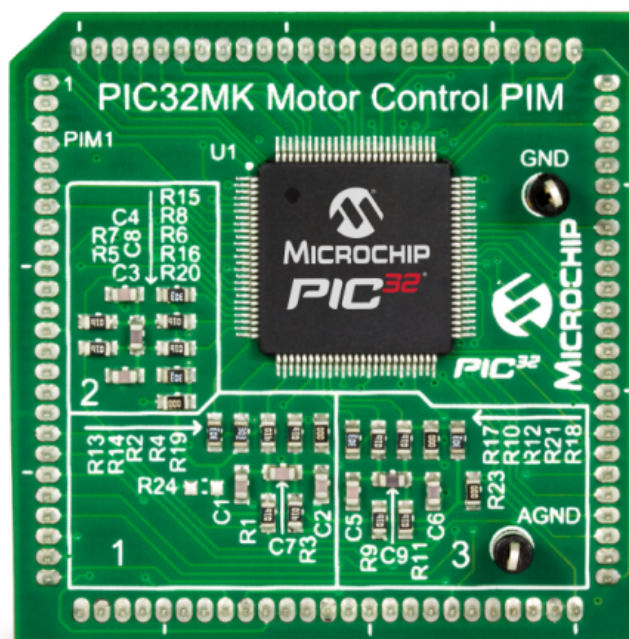
Microchip Part Number:

MA320024

Product Page:

<http://www.microchip.com/developmenttools/productdetails.aspx?partno=ma320024>

Hardware Layout



PIC32MX XLP Starter Kit

Provides information on the PIC32MX XLP Starter Kit.

Description

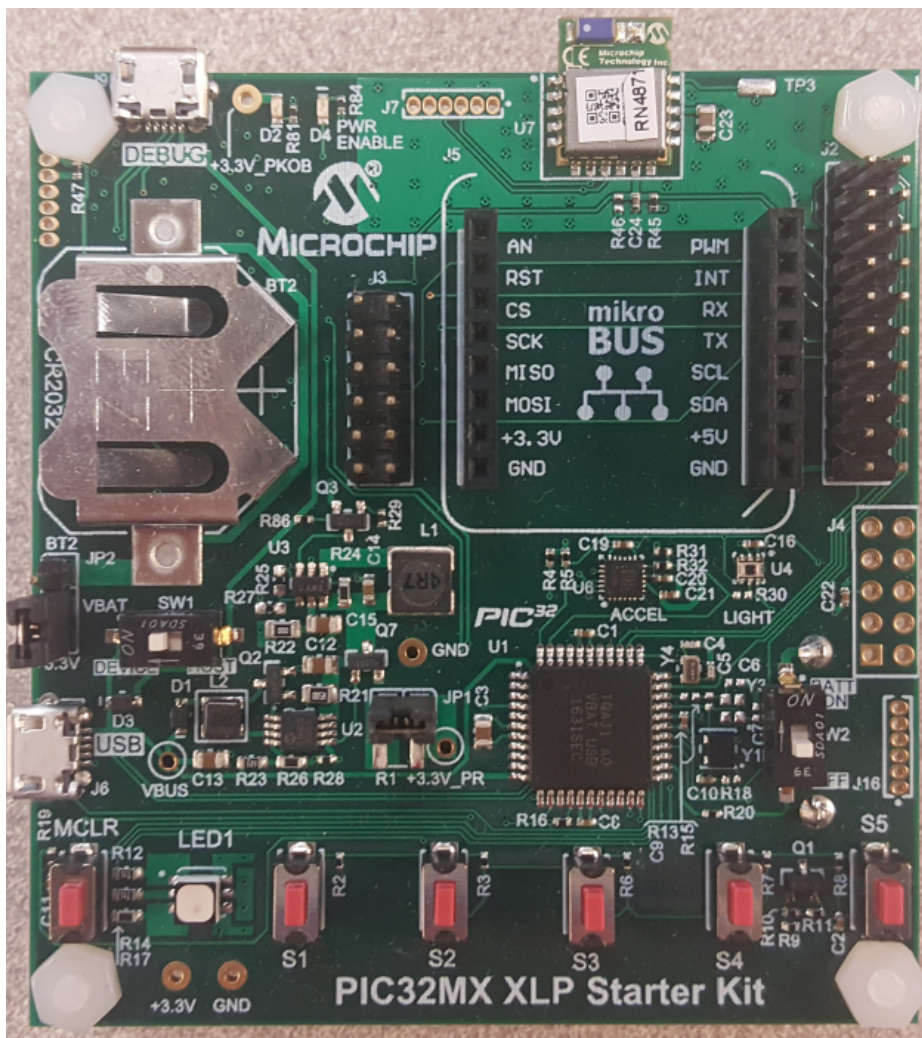
Microchip Part Number:

DM320105

Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320105>

Hardware Layout



PIC32MX1/2/5 Starter Kit

Provides information on the PIC32MX1/2/5 Starter Kit.

Description

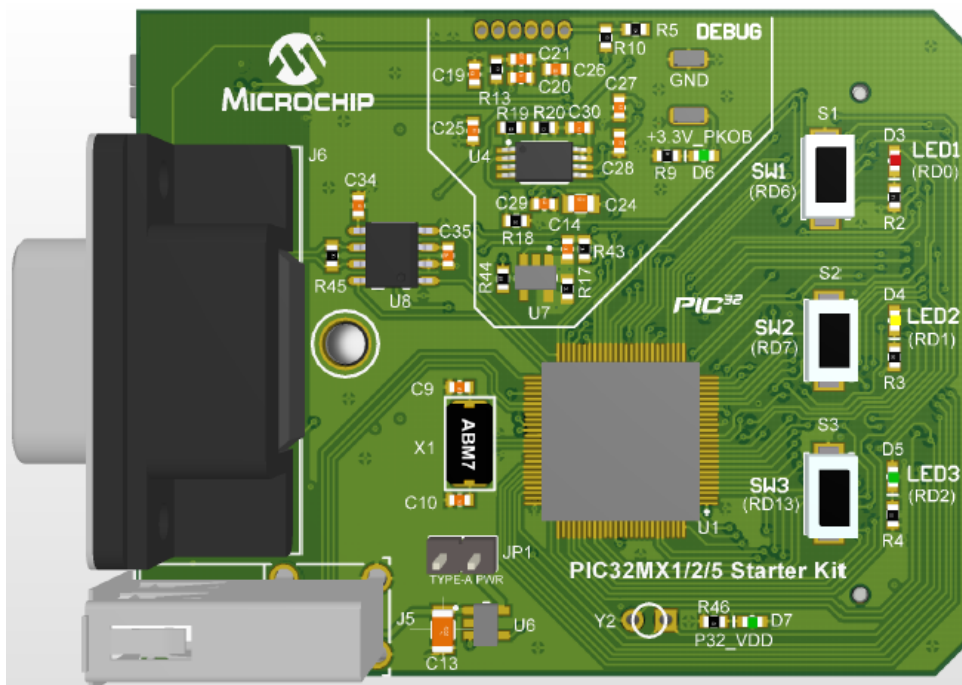
Microchip Part Number:

DM320100

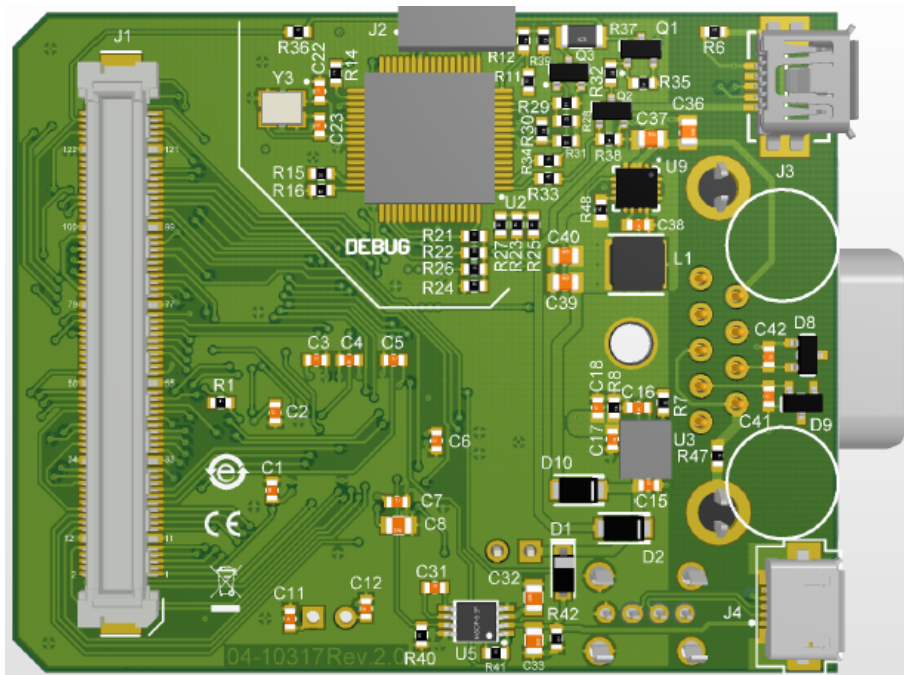
Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320100>

Hardware Layout (Front)



Hardware Layout (Back)



PIC32MX270F256D Plug-in Module (PIM)

Provides information on the PIC32MX270F256D Plug-in Module (PIM).

Description

Two PIC32MX270F256D PIMs are available, one for use with the PIC32 Bluetooth Audio Development Kit, and one for use with the Explorer 16 Development Board.

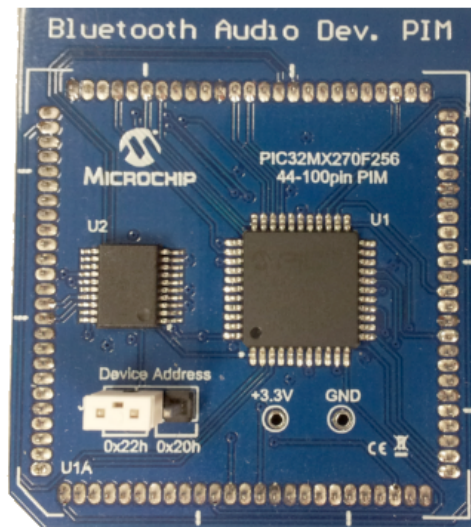
Microchip Part Number:

MA320013

This PIM is required while using the [PIC32 Bluetooth Audio Development Kit](#)

microchipDIRECT Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=MA320013>

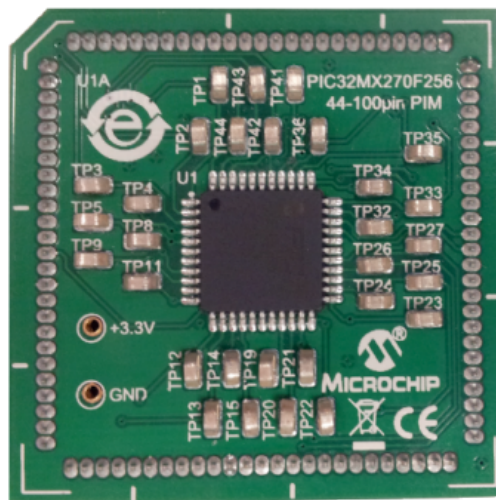
Hardware Layout**Microchip Part Number:**

MA320014

This PIM is required while using the [Explorer 16 Development Board](#).

microchipDIRECT Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=MA320014>

**PIC32MX270F512L Plug-in Module (PIM)**

Provides information on the PIC32MX270F512L Plug-in Module (PIM).

Description**Microchip Part Number:**

MA320017

This PIM is for use with the [PIC32 Bluetooth Audio Development Kit](#).

microchipDIRECT Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=MA320017>

Hardware Layout



PIC32MX360F512L Plug-in Module (PIM)

Provides information on the PIC32MX360F512L Plug-in Module (PIM).

Description

Microchip Part Number:

MA320001

microchipDIRECT Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=MA320001>

Hardware Layout



PIC32MX460F512L Plug-in Module (PIM)

Provides information on the PIC32MX460F512L Plug-in Module (PIM).

Description

Microchip Part Number:

MA320002

microchipDIRECT Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=MA320002>

Hardware Layout



PIC32MX450/470F512L Plug-in Module (PIM)

Provides information on the PIC32MX450/470F512L Plug-in Module (PIM).

Description

Microchip Part Number:

MA320002-2

microchipDIRECT Product Page:

<http://www.microchipdirect.com/ProductSearch.aspx?Keywords=MA320002-2>

Hardware Layout



PIC32MX470 Curiosity Development Board

Provides information on the PIC32MX470 Curiosity Development Board.

Description

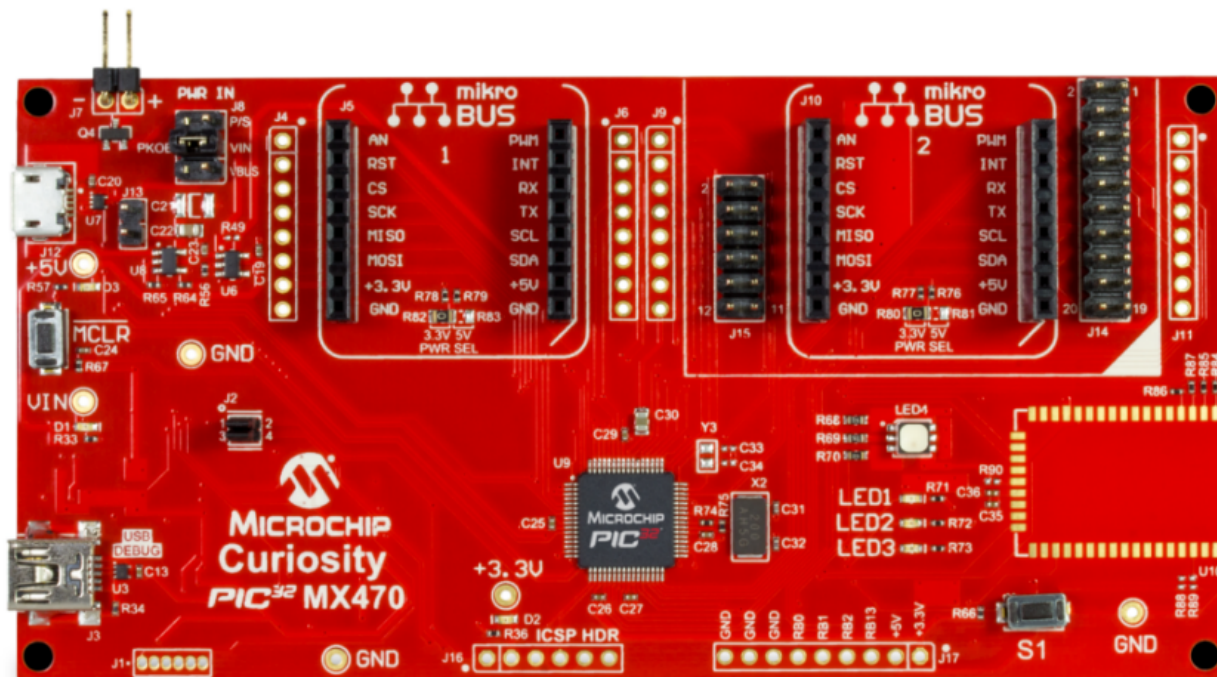
Microchip Part Numbers:

DM320103

Microchip Product Pages:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320103>

Hardware Layout



PIC32MX570F512L Plug-in Module (PIM)

Provides information on the PIC32MX570F512L Plug-in Module (PIM).

Description

Microchip Part Number:

MA320015

microchipDIRECT Product Page:

<http://www.microchipdirect.com/ProductSearch.aspx?Keywords=MA320015>

Hardware Layout



PIC32MX795F512L CAN-USB Plug-in Module (PIM)

Provides information on the PIC32MX795F512L CAN-USB Plug-in Module (PIM).

Description

Microchip Part Number:

MA320003

microchipDIRECT Product Page:

<http://www.microchipdirect.com/productsearch.aspx?keywords=MA320003>

**PIC32MZ2048EFH100 Plug-in Module (PIM)**

Provides information on the PIC32MZ2048EFH100 Plug-in Module (PIM).

Description**Microchip Part Number:**

MA320012

microchipDIRECT Product Page:

<http://www.microchipdirect.com/productsearch.aspx?keywords=MA320012>

Hardware Layout**PIC32MZ2048EFH100 Plug-in Module (PIM)**

Provides information on the PIC32MZ2048EFH100 Plug-in Module (PIM).

Description**Microchip Part Number:**

MA320019

Microchip Product Page:

<http://www.microchip.com>

Hardware Layout



PIC32MZ2048EFH144 Audio Plug-in Module (PIM)

Provides information on the PIC32MZ2048EFH144 Audio Plug-in Module (PIM).

Description

Microchip Part Number:

MA320016

microchipDIRECT Product Page:

<http://www.microchipdirect.com/productsearch.aspx?keywords=MA320016>

Hardware Layout



PIC32MZ2048EFH144 Audio Plug-in Module (PIM)

Provides information on the PIC32MZ2048EFH144 Audio Plug-in Module (PIM).

Description

Microchip Part Number:

MA320018

microchipDIRECT Product Page:

<http://www.microchipdirect.com/productsearch.aspx?keywords=MA320018>

Hardware Layout



PIC32MZ Audio PIM

Provides information on the PIC32MZ Audio PIM.

Description

Microchip Part Number:

MA320016

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=MA320016>

Hardware Layout



PIC32MZ EF Curiosity Development Board

Provides information on the PIC32MZ EF Curiosity Development Board.

Description

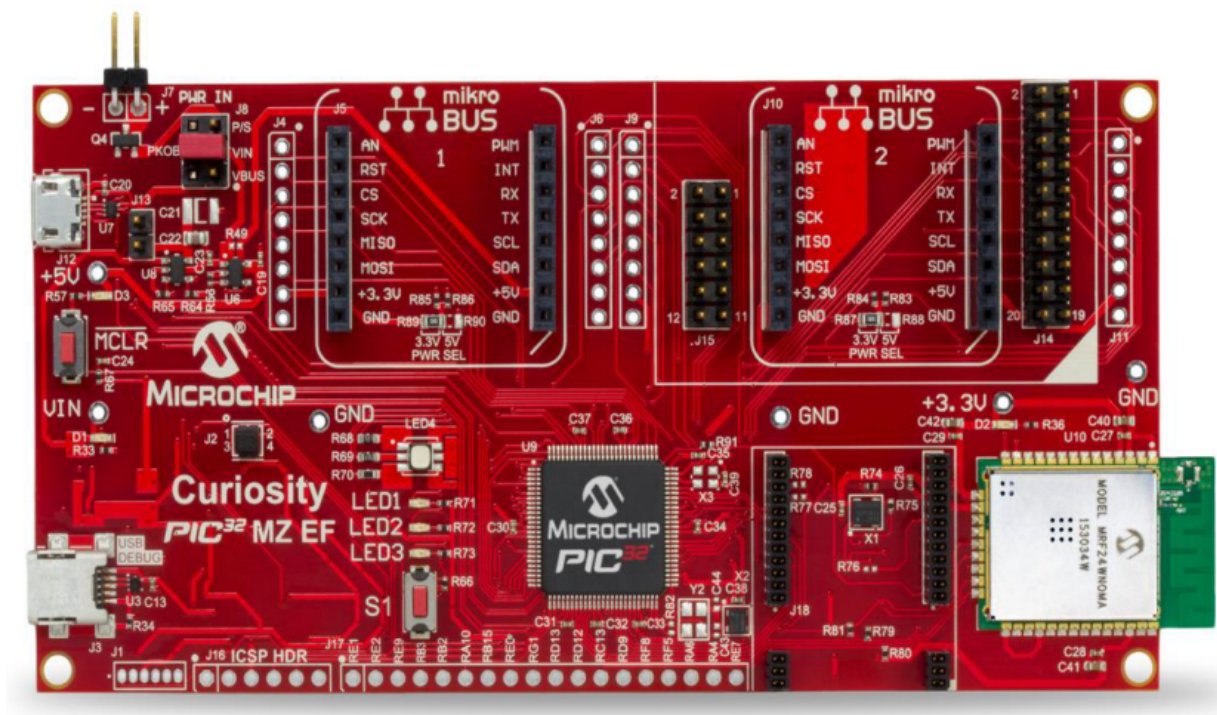
Microchip Part Numbers:

DM320104

Microchip Product Pages:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320104>

Hardware Layout



PIC32MZ Embedded Connectivity (EC) Starter Kit

Provides information on the PIC32MZ Embedded Connectivity (EC) Starter Kit.

Description

Microchip Part Numbers:

DM320006

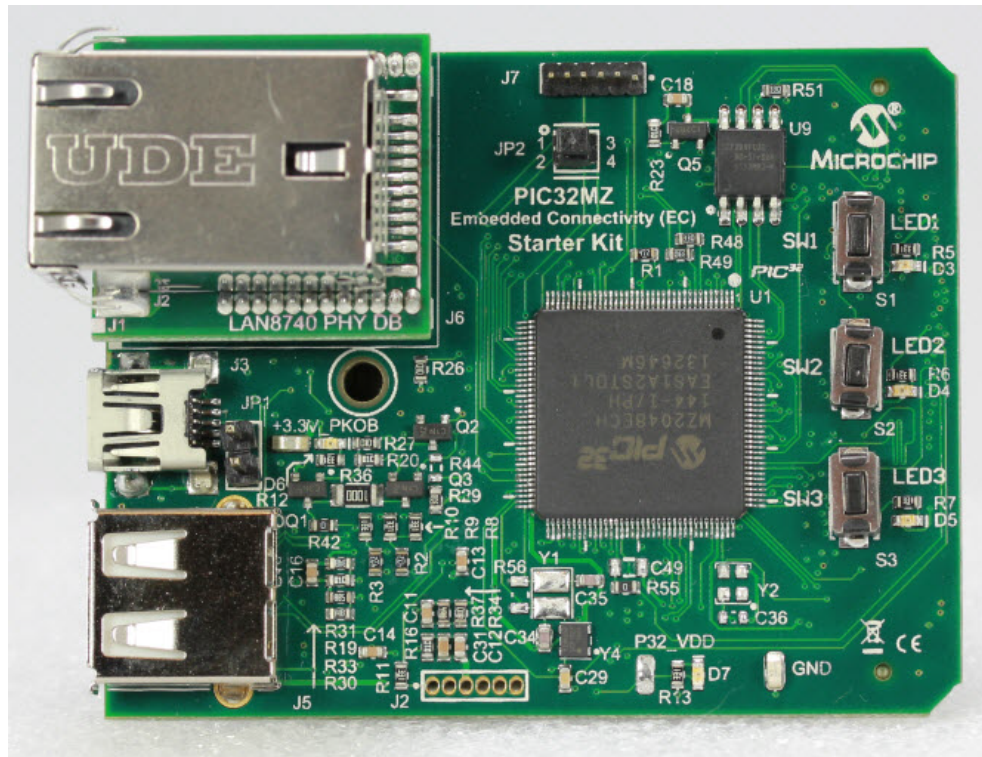
DM320006-C (with Crypto Engine)

Microchip Product Pages:

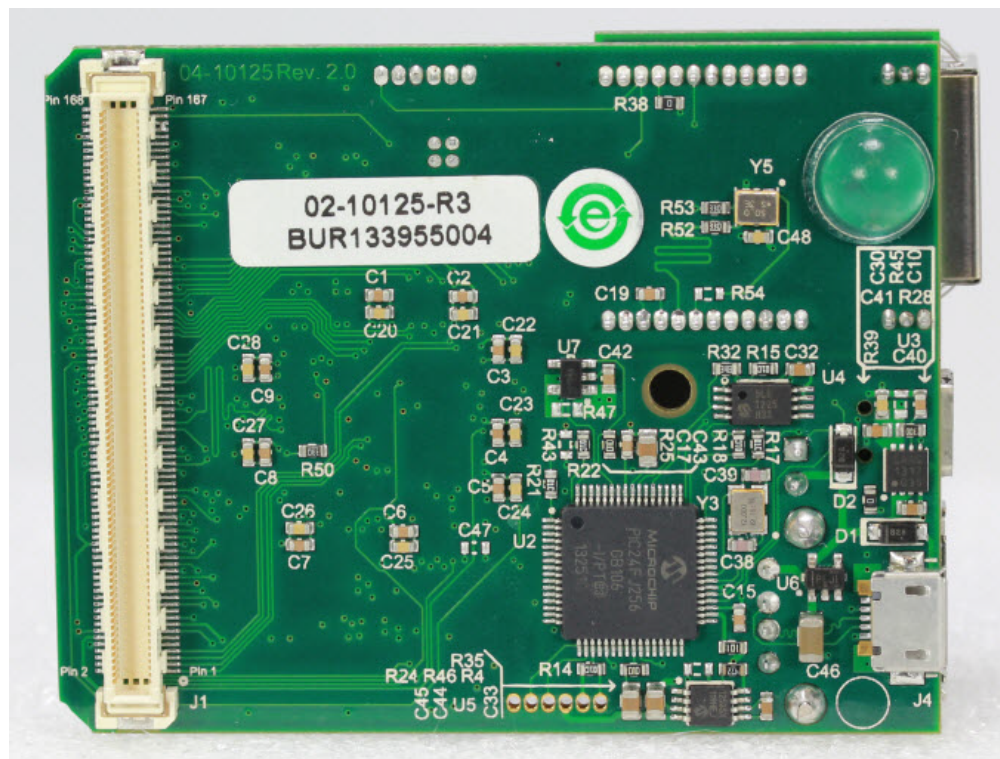
<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320006>

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320006-C>

Hardware Layout (Front)



Hardware Layout (Back)



PIC32MZ Embedded Connectivity with Floating Point Unit (EF) Starter Kit

Provides information on the PIC32MZ Embedded Connectivity with Floating Point Unit (EF) Starter Kit.

Description

Microchip Part Numbers:

DM320007 (without Crypto)

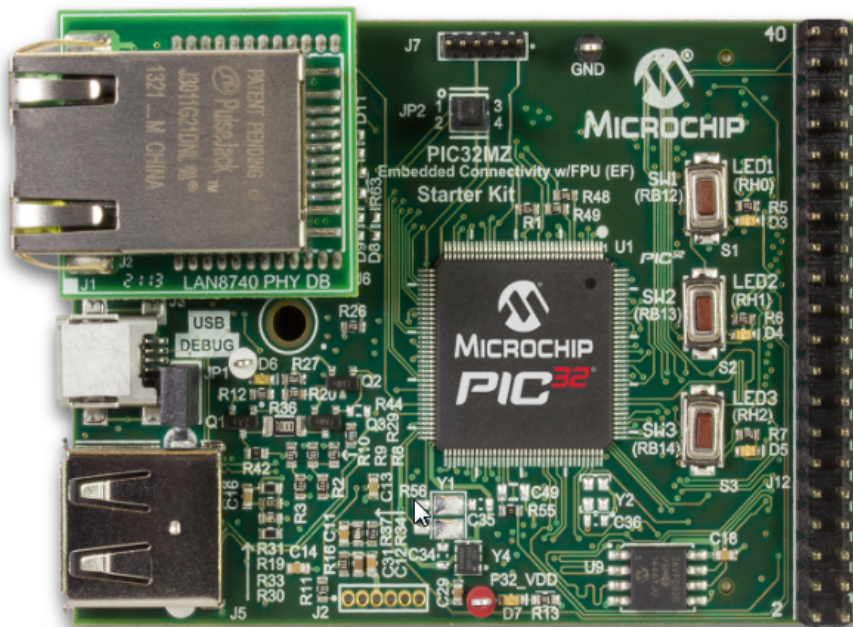
DM320007-C (with Crypto)

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320007>

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320007-C>

Hardware Layout



Note:

The hardware layout is subject to change and may differ from the photograph provided.

PIC32MZ Embedded Graphics with Disabled DRAM (DA) Starter Kit

Provides information about the PIC32MZ Embedded Graphics with Disabled DRAM (DA) Starter Kit.

Description

Microchip Part Number:



Note:

Please contact your local Microchip sales office for information on obtaining this board.

PIC32MZ Embedded Graphics with External DRAM (DA) Starter Kit

Provides information on the PIC32MZ Embedded Graphics with External DRAM (DA) Starter Kit.

Description

Microchip Part Numbers:

DM320008 (without Crypto)

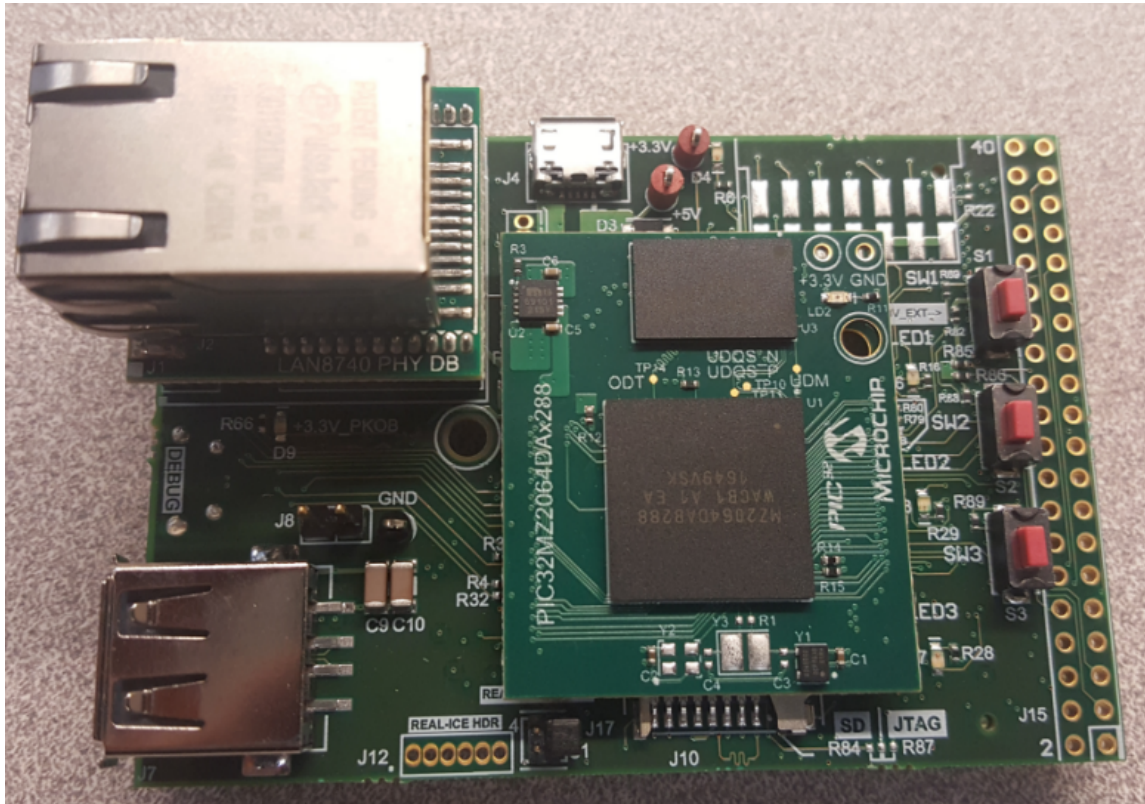
DM320008-C (with Crypto)

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320008>

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320008-C>

Hardware Layout



PIC32MZ Embedded Graphics with Internal DRAM (DA) Starter Kit

Provides information on the PIC32MZ Embedded Graphics with Internal DRAM (DA) Starter Kit.

Description

Microchip Part Numbers:

DM320010 (without Crypto)

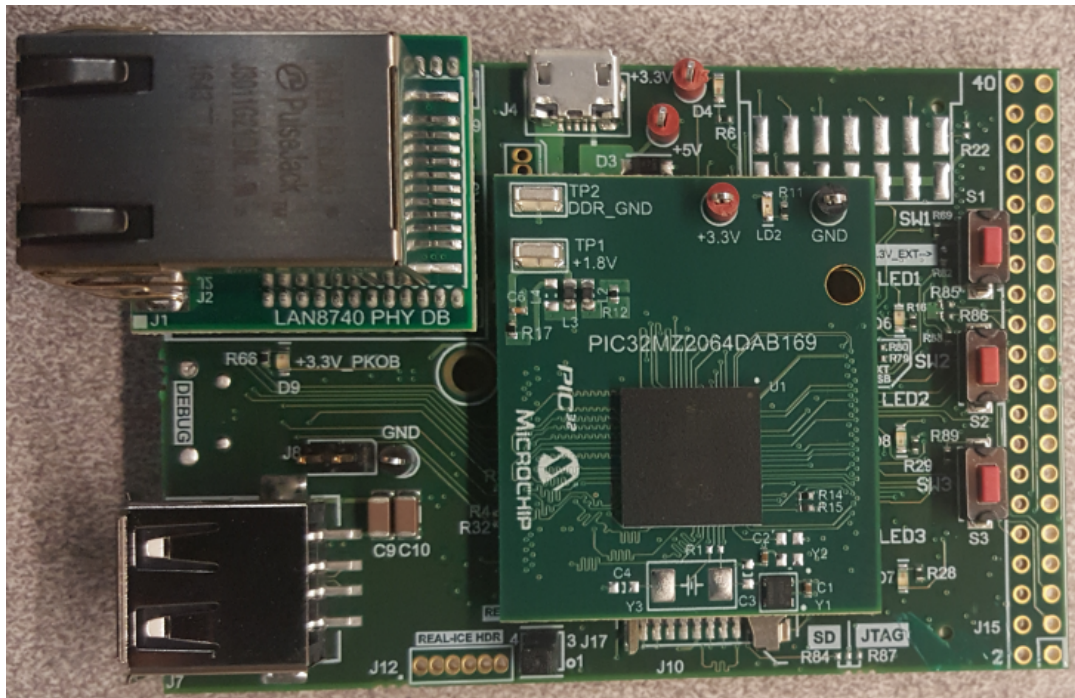
DM320010-C (with Crypto)

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320010>

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=DM320010-C>

Hardware Layout



PIC32MZ Starter Kit Adapter Board

Provides information on the PIC32MZ Starter Kit Adapter Board.

Description

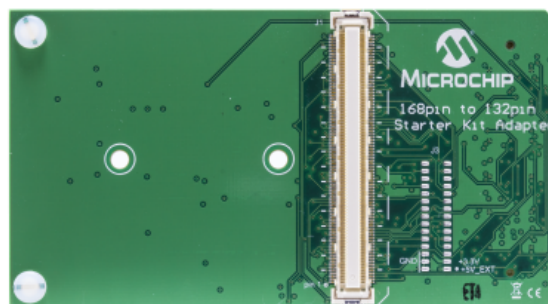
Microchip Part Number:

AC320006

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC320006>

Hardware Layout



PIC32WK Wi-Fi Starter Kit

Provides information on the PIC32WK Wi-Fi Starter Kit.

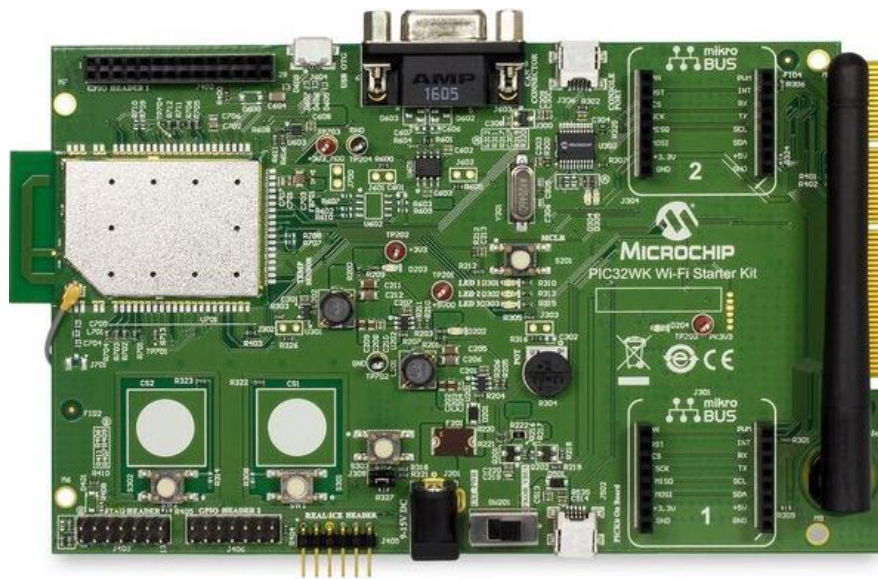
Description

Microchip Part Numbers:

DM320102-1

**Note:**

Please contact your local Microchip sales office for information on obtaining this board.

Hardware Layout**PICkit 3 In-Circuit Debugger**

Provides information on the PICkit™ 3 In-Circuit Debugger.

Description

Microchip Part Number:

PG164130

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=PG164130>

Hardware Layout



PICtail Daughter Board for SD and MMC

Provides information on the PICtail Daughter Board for SD and MMC.

Description

Microchip Part Number:

AC164122

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC164122>

Hardware Layout



PICtail Plus Expansion Board

Provides information on the PICtail Plus Expansion Board

Description

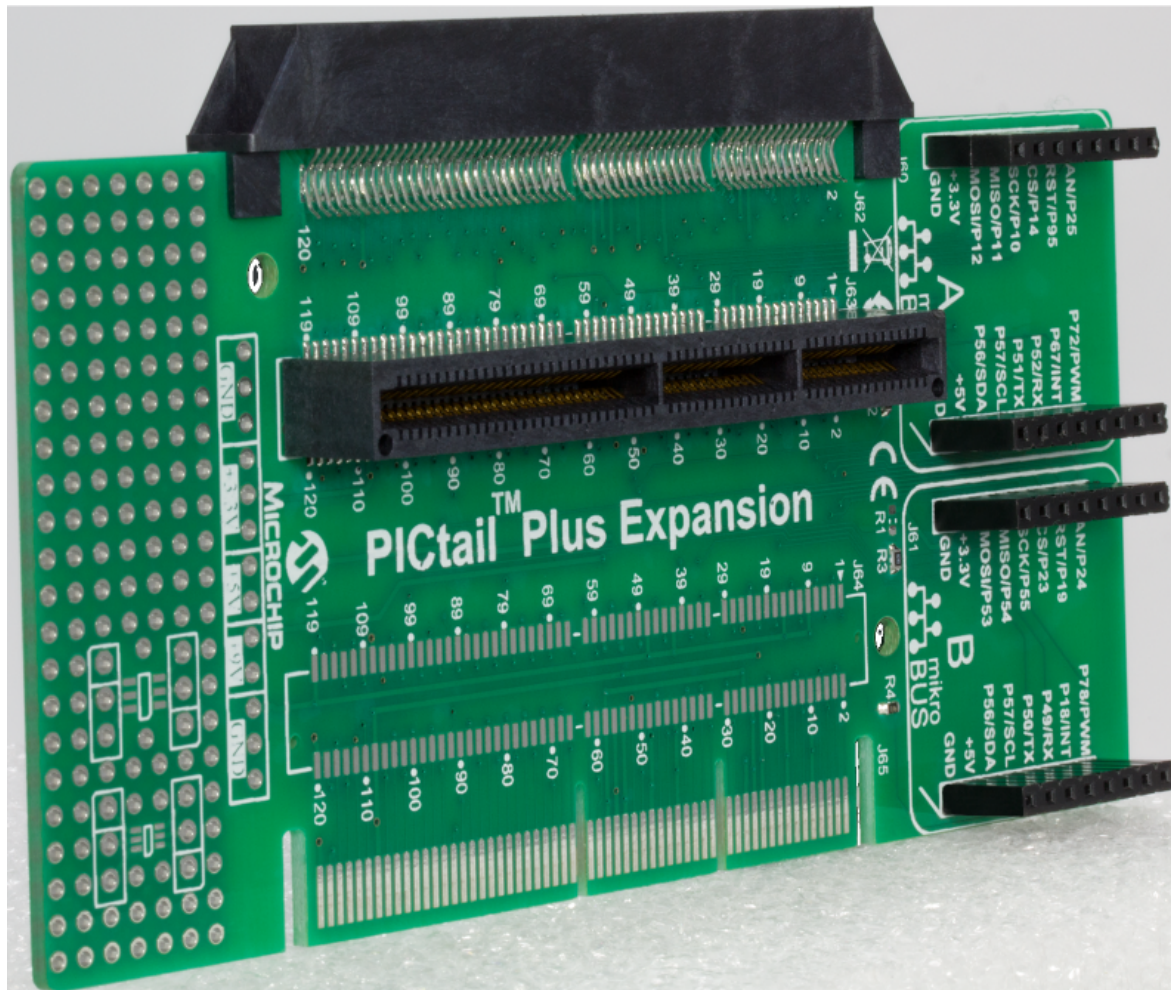
Microchip Part Number:

AC240100

Microchip Product Page:

<http://www.microchip.com/developmenttools/productdetails.aspx?partno=ac240100>

Hardware Layout



Prototype PICtail Plus Daughter Board

Provides information on the Prototype PICtail Plus Daughter Board.

Description

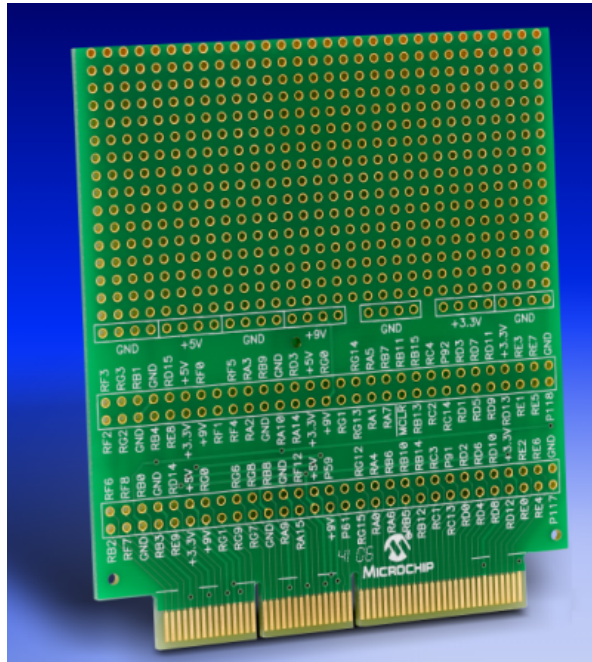
Microchip Part Number:

AC164126

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC164126>

Hardware Layout



SAM V71 Xplained Ultra Evaluation Kit

Provides information on the SAM V71 Xplained Ultra Evaluation Kit.

Description

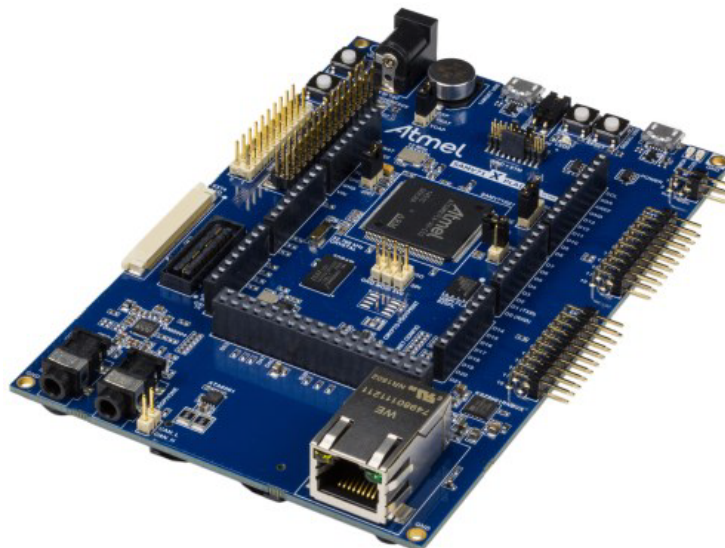
Microchip Part Number:

ATSAMV71-XULT

Microchip Product Page:

<http://www.microchip.com/developmenttools/productdetails.aspx?partno=atsamv71-xult>

Hardware Layout



Starter Kit I/O Expansion Board

Provides information on the Starter Kit I/O Expansion Board.

Description

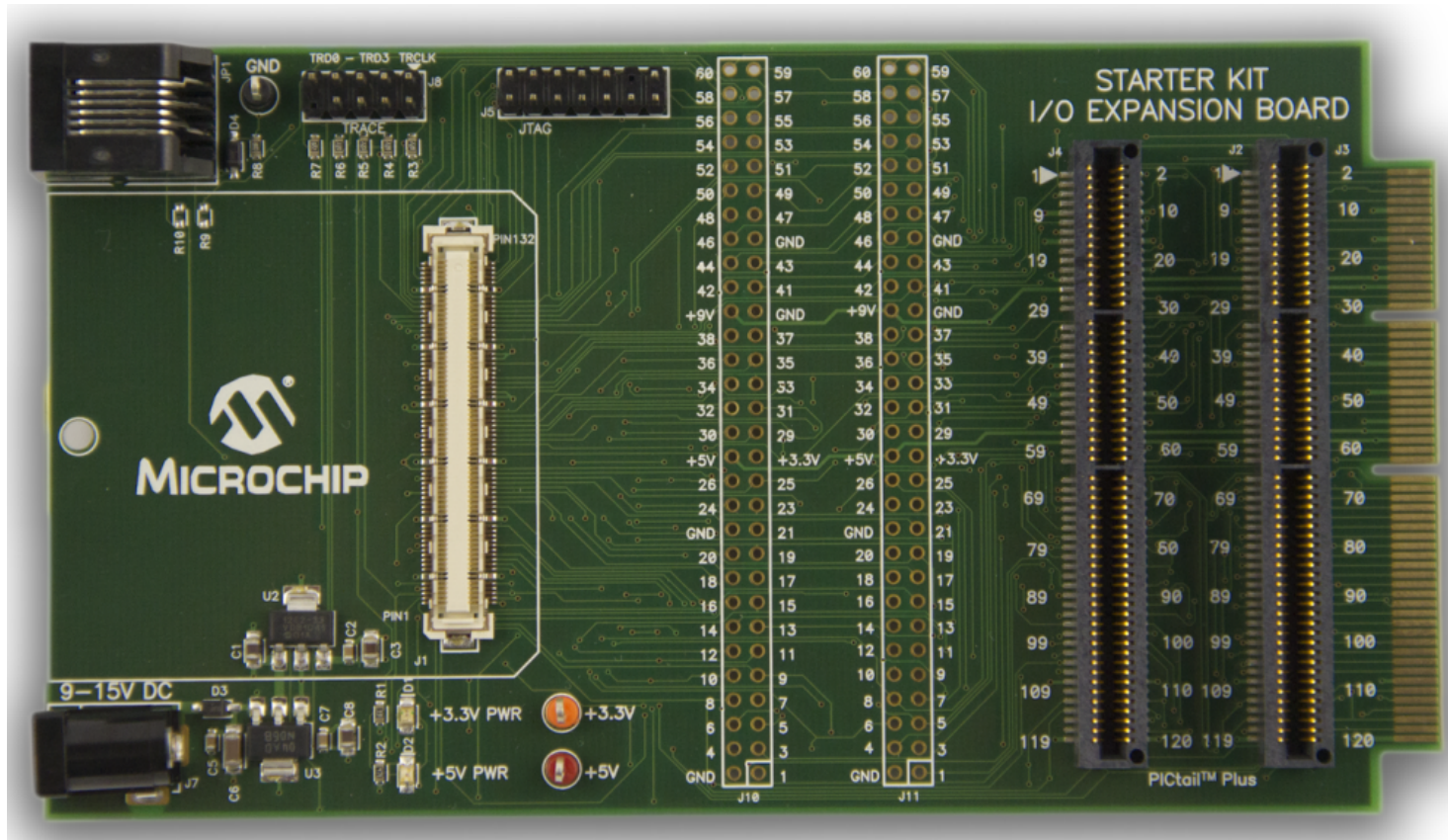
Microchip Part Number:

DM320002

Microchip Product Page:

<http://www.microchip.com/DevelopmentTools/ProductDetails.aspx?PartNO=DM320002>

Hardware Layout



USB PICtail Plus Daughter Board

Provides information on the USB PICtail Plus Daughter Board.

Description

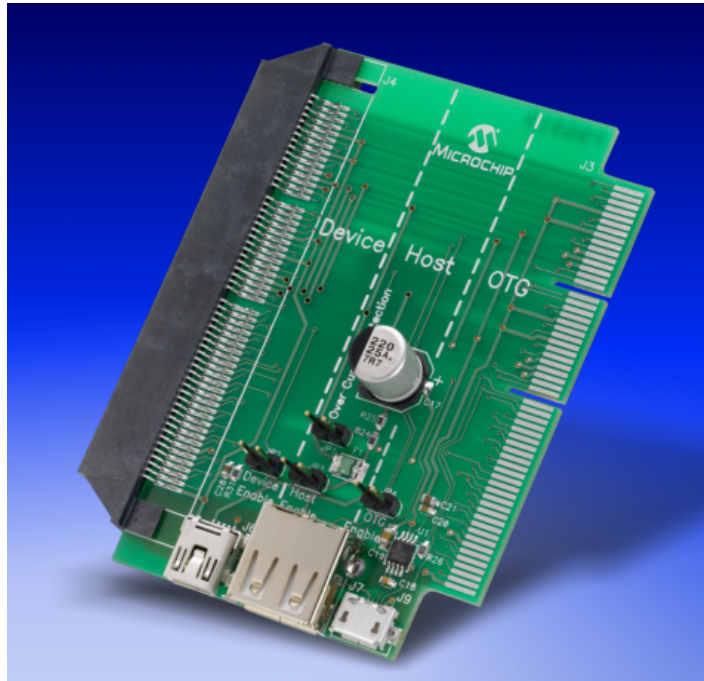
Microchip Part Number:

AC164131

Microchip Product Page:

<http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC164131>

Hardware Layout



WINC1500 Wi-Fi module

Provides information on the WINC1500 Wi-Fi Module

Description

Microchip Part Number:

ATWINC1500-MR210PB1952

Microchip Product Page:

<http://www.microchip.com/wwwproducts/en/atwinc1500>

Hardware Layout



WILC1000 Wi-Fi PICtail/PICtail Plus Daughter Board

Provides information about the WILC1000 Wi-Fi PICtail/PICtail Plus Daughter Board.

Description

Microchip Part Number:

**Note:**

Please contact your local Microchip sales office for information on obtaining this board.

WINC1500 Wi-Fi PICtail/PICtail Plus Daughter Board

Provides information about the WINC1500 Wi-Fi PICtail/PICtail Plus Daughter Board.

Description

Microchip Part Number:

AC164156

Microchip Product Page:

<http://www.microchip.com/developmenttools/productdetails.aspx?partno=ac164156>

Hardware Layout



Board Support Packages Help

This section describes the Board Support Packages (BSP) that are available in MPLAB Harmony.

Introduction

This topic provides information for the Board Support Package (BSP) capabilities MPLAB Harmony.

Description

A MPLAB Harmony Board Support Package (BSP) provides support for board-specific hardware, which is provided in two forms:

- **BSP Library:** Provides board-specific initialization code and an interface to switches, LEDs, and GPIO pins
- **Initial Configuration:** Provides a starting point for configuration settings for clocks, ports, and selected libraries when creating a new project (or project configuration)

BSP Library

A MPLAB Harmony BSP library (consisting of a `bsp.h` and `bsp.c` file) is configured specifically for the selected board and generated into the project's configuration folder at the following location:

```
<project>/firmware/src/system_config/<configuration>/bsp
```

Where:

`<project>` - Is the root folder of the project

`<configuration>` - Is the project's configuration folder when the BSP is generated

A BSP library has a set of system interface functions and a set of client interface functions. However, unlike most MPLAB Harmony library modules, BSPs do not often have a state machine of their own, and therefore, they usually do not have a `BSP_Tasks` function or other system interface functions. In a project configuration, the BSP's initialize function (`BSP_Initialize`) is normally called from the system's `SYS_Initialize` function before any other initialization function (except those that initialize any required core processor capabilities). This allows the BSP to initialize any necessary board-specific hardware (like memory and GPIO controllers) before other libraries are initialized. Refer to *Volume IV: MPLAB Harmony Development > MPLAB Harmony Driver Development Guide > System Interface* for additional information on system interface functions.)

A BSP also provides a simple and extensible client interface for controlling switches, LEDs, and other GPIO pins. This interface includes functions to turn LEDs and pins on-or-off and to toggle, set, or get their current state, where so configured. The names of these functions all begin with "`BSP_LED`" and are followed by the operation verb (On, Off, Toggle, StateSet, or StateGet). There is also a BSP function to get the current state of switches on the board called `BSP_SwitchStateGet`. These functions accept values from a custom-generated enumeration as a parameter to identify the specific switches, LEDs, and pins on which to perform the operation. The names given to the switches, LEDs, and other pins used by the BSP interface functions are defined in the MPLAB Harmony Configurator's Pin Settings window. These names are used to define the labels in the custom-generated enumerations used by these BSP functions. Refer to *Volume III: MPLAB Harmony Configurator (MHC) > MPLAB Harmony Configurator User's Guide > MPLAB Harmony Graphical Pin Manager* and to the following topics in this section for additional information on how to define custom pin names.

Additionally, the BSP client interface is extensible. The names given to the switches, LEDs, and pins in the Graphical Pin Manager are also prepended to the operation verbs (On, Off, Toggle, StateSet, and StateGet) to generate a completely custom set of control functions for each pin. These functions are described in the Library Interface section of this document using monikers prefixed with "`custom_gpio_name`", "`custom_switch_name`", and "`custom_gpio_name`". No actual functions exist with these names. Instead, these labels are replaced with the custom name given to each pin and function-like macros are generated (defined in `system_config.h`) to provide functions that directly access these IO pins. These macros can be used by the application, but be aware that the common BSP functions may be more portable as these custom macros must be generated by the configuration. For more information, refer to [Extending the BSP Library Interface](#).

Initial Configuration

An initial BSP configuration is basically another type of MHC configuration file. (Refer to *Volume III: MPLAB Harmony Configurator (MHC) > MPLAB Harmony Configurator Developer's Guide > MHC Files* for details of the MHC file.) The initial configuration can contain settings selected using the MHC (see *Volume III: MPLAB Harmony Configurator (MHC)* for information on using the MHC). When a new project (or a new configuration of an existing project) is created (or any time a BSP configuration is imported), the MHC allows the user the option to select an initial board configuration (MHC) file. If one is selected, the MHC initializes the project configuration with the settings in the selected file (overwriting the default or existing settings). This gives the user a known starting point from which to configure the project, based upon the board selected. A single BSP can have multiple initial board configuration files with different configuration settings, allowing the user to select the initial configuration that most closely represents the desired project configuration.

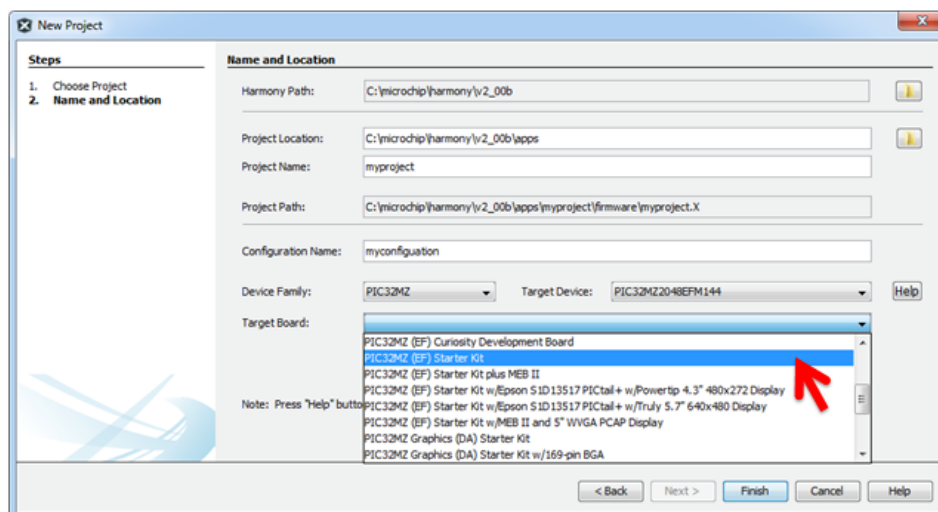
Using a BSP

This section provides information on using a MPLAB Harmony BSP.

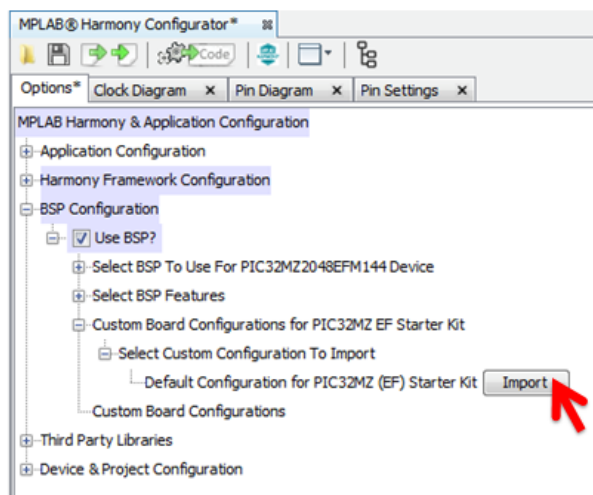
Description

To use a BSP, select the desired BSP from the list of available BSPs in the MPLAB X IDE New Project Wizard when creating a new project, as

shown in the following figure, and complete the New Project Wizard.



The default board configuration settings will have been applied when the project is created, even before clicking the Generate Code button. You may then make modifications to the current configuration as desired, or load a custom board configuration, or reload the default BSP configuration by expanding *BSP Configuration > USB BSP > Custom Board Configurations for [BSP Name] > Select Custom Configuration to Import*, and then clicking the **Import** button as shown in the following figure. The imported settings will overwrite any current settings, and then be saved in the configuration's MHC file (not the BSP's board configuration file) when you click either the **Generate Code** or **Save Configuration** button.



Refer to *Volume III: MPLAB Harmony Configurator (MHC) > MPLAB Harmony Configurator User's Guide* for information on using the MHC. To utilize the BSP library functions, simply call the desired function from your application as shown in the example section within the documentation for each function in the Library Interface section of this document.

Customizing a BSP

Describes how to customize a BSP library and create custom initial board configurations.

Description

Customizations can be made by extending or modifying the BSP library source code or by creating custom initial board configurations, as described in the following topics.

Extending the BSP Library Interface

Describes how to extend the BSP library interface.

Description

The MHC will generate custom code that extends the BSP API when any of the following pin functions are selected in the Pin Settings table.

BSP-Related Pin Functions

- LED_AH Selects active high LED functionality that will generate custom named macros to control an LED that turns on when the pin is driven

high

- LED_AL Selects active low LED functionality that will generate custom named macros to control an LED that turns on when the pin is driven low
- GPIO Selects general purpose IO pin functionality that will generate custom named macro constants for use with the Ports System Service functions
- GPIO_IN Selects general purpose input pin that will generate a custom named macro to obtain the state of a general purpose input pin
- GPIO_OUT Selects general purpose output pin functionality that will generate custom named macros to control a general purpose output pin

Select one of the BSP-related pin functions in the MHC's Pin Setting window and provide a custom name for the pin, as shown in the following figure.

Pin Number	Pin ID	Voltage Tolerance	Name	Function	Direction (TRIS)	Latch (LAT)	Open Drain (ODC)	Mode (ANSEL)	Change Notification (CNEN)
1	RG15		CUSTOM_LED	LED_AH	Out	Low	<input type="checkbox"/>	Digital	<input type="checkbox"/>
2	RA5		CUSTOM_SWITCH	SWITCH	In	Low	<input type="checkbox"/>	Digital	<input checked="" type="checkbox"/>
3	RE5		CUSTOM_GPIO	GPIO	In	Low	<input type="checkbox"/>	Digital	<input type="checkbox"/>
4	RE6		CUSTOM_GPIO_IN	GPIO_IN	In	Low	<input type="checkbox"/>	Digital	<input type="checkbox"/>
5	RE7		CUSTOM_GPIO_OUT	GPIO_OUT	Out	Low	<input type="checkbox"/>	Digital	<input type="checkbox"/>



Note:

Be sure to correctly configure the pin direction, initial latch value, digital-or-analog mode, and other settings.

After clicking the **Generate Code** button, the `bsp.h` file will be updated to contain macro definitions for any custom-named pins that are configured for BSP-related pin functions, as shown in the following examples.

Example Macros for LED_AH or LED_AL Pin Named CUSTOM_LED

```
#ifndef CUSTOM_LEDToggle
#define CUSTOM_LEDToggle() PLIB_PORTS_PinToggle \
    (PORTS_ID_0, PORT_CHANNEL_G, PORTS_BIT_POS_15)
#endif
#ifndef CUSTOM_LEDon
#define CUSTOM_LEDon() PLIB_PORTS_PinSet \
    (PORTS_ID_0, PORT_CHANNEL_G, PORTS_BIT_POS_15)
#endif
#ifndef CUSTOM_LEDOff
#define CUSTOM_LEDOff() PLIB_PORTS_PinClear \
    (PORTS_ID_0, PORT_CHANNEL_G, PORTS_BIT_POS_15)
#endif
#ifndef CUSTOM_LEDStateGet
#define CUSTOM_LEDStateGet() PLIB_PORTS_PinGetLatched \
    (PORTS_ID_0, PORT_CHANNEL_G, PORTS_BIT_POS_15)
#endif
```



Note:

The macros will be implemented to correctly set or clear the pin based on the selection of active-high or active-low LED function.

Example Macro for SWITCH Pin Named CUSTOM_SWITCH

```
#ifndef CUSTOM_SWITCHStateGet
#define CUSTOM_SWITCHStateGet() PLIB_PORTS_PinGet \
    (PORTS_ID_0, PORT_CHANNEL_A, PORTS_BIT_POS_5)
#endif
```

Example Macros for GPIO_OUT Pin Named CUSTOM_GPIO_OUT

```
#ifndef CUSTOM_GPIO_OUTToggle
#define CUSTOM_GPIO_OUTToggle() PLIB_PORTS_PinToggle \
    (PORTS_ID_0, PORT_CHANNEL_E, PORTS_BIT_POS_7)
#endif
#ifndef CUSTOM_GPIO_OUTOn
#define CUSTOM_GPIO_OUTOn() PLIB_PORTS_PinSet \
    (PORTS_ID_0, PORT_CHANNEL_E, PORTS_BIT_POS_7)
#endif
#ifndef CUSTOM_GPIO_OUTOff
#define CUSTOM_GPIO_OUTOff() PLIB_PORTS_PinClear \
    (PORTS_ID_0, PORT_CHANNEL_E, PORTS_BIT_POS_7)
#endif
```

```
#ifndef CUSTOM_GPIO_OUTStateGet
#define CUSTOM_GPIO_OUTStateGet() PLIB_PORTS_PinGetLatched \
(PORTS_ID_0, PORT_CHANNEL_E, PORTS_BIT_POS_7)
#endif
#ifndef CUSTOM_GPIO_OUTStateSet
#define CUSTOM_GPIO_OUTStateSet(Value) PLIB_PORTS_PinWrite \
(PORTS_ID_0, PORT_CHANNEL_E, PORTS_BIT_POS_7, Value)
#endif
```

Example Macro for GPIO_IN Pin Named CUSTOM_GPIO_IN

```
#ifndef CUSTOM_GPIO_INStateGet
#define CUSTOM_GPIO_INStateGet() PLIB_PORTS_PinGet \
(PORTS_ID_0, PORT_CHANNEL_E, PORTS_BIT_POS_6)
#endif
```

Example Macros for GPIO Pin Named CUSTOM_GPIO

```
#define CUSTOM_GPIO_PORT PORT_CHANNEL_E
#define CUSTOM_GPIO_PIN PORTS_BIT_POS_5
#define CUSTOM_GPIO_PIN_MASK (0x1 << 5)
```



Note:

Custom GPIO pin name macros are intended for use with the PORTS PLIB or System Service library interface functions. Refer to *Volume IV: MPLAB Harmony Framework Reference > Peripheral Libraries Help > Ports Peripheral Library* or *Volume IV: MPLAB Harmony Framework Reference > System Service Libraries Help > Ports System Service Library* for information on these functions.

The extended BSP interface macros can be overridden by defining them in the project's `system_config.h` header, if desired.



Note:

Current versions of MPLAB Harmony only define these macros in the `system_config.h` file, so they may be overridden by redefining them there.

Modifying the BSP Library Source Code

Describes how to modify the BSP library source code.

Description

The `bsp.h` and `bsp.c` source code files are generated by the MHC directly into the project's current configuration folder as described in the [Introduction](#). Therefore, these files can be directly edited as required to support the configuration. The MHC will show a “diff” tool window when generating updates to the configuration, allowing you to merge the newly generated code with the existing code.

Creating Custom Initial Board Configurations

Describes how to create custom initial board configurations.

Description

An initial board configuration is a MHC (*.mhc) file that has the same format and contains the same sort of information (about configuration selections made using the MHC graphical user interface) as the MHC file in which a project configuration is stored. The difference is that it has been added to the MHC's list of initial board configurations for an associated BSP library within an installation of MPLAB Harmony.

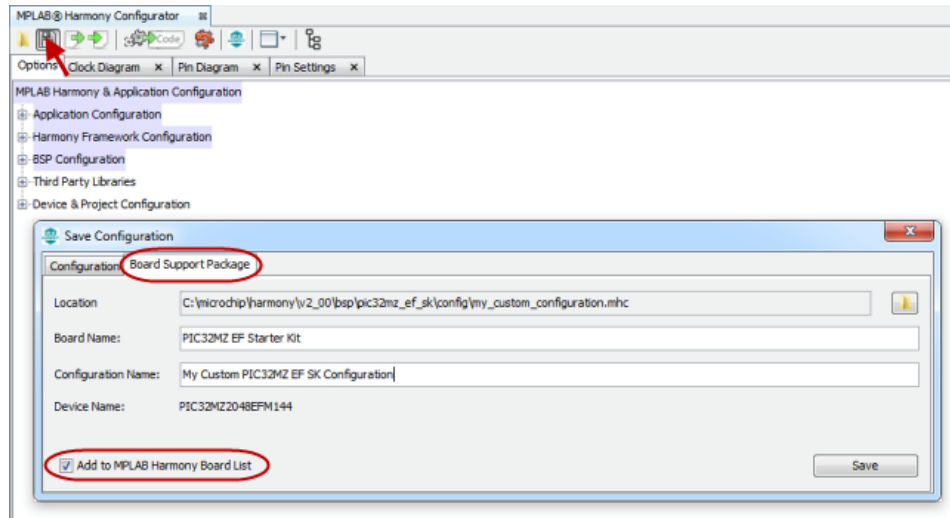
The easiest way to create a custom initial board configuration is to create a temporary MPLAB Harmony project based upon an existing BSP, and import and customize any desired configuration settings from existing project configurations. Refer to *Volume III: MPLAB Harmony Configurator (MHC) > MPLAB Harmony Configurator User's Guide* for information on how to create new projects using the MHC. Refer to *Volume III: MPLAB Harmony Configurator (MHC) > MPLAB Harmony Configurator User's Guide > Importing and Exporting Data* for information on how to import settings from other configurations.



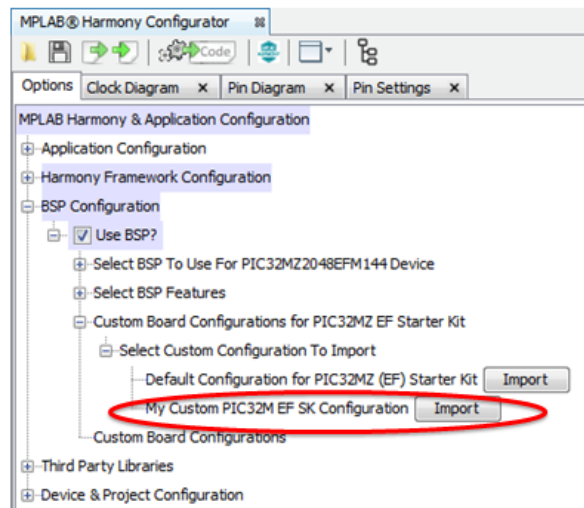
Note:

While it is possible to import or select application settings in an initial board configuration, it is not usually recommended, as the file may be used to initialize configurations for various different applications.

Once you have imported and/or selected all of the desired settings, save the configuration as an initial board configuration by choosing the Board Support Package tab, instead of the default Configurations tab, as shown in the following figure.



To associate the initial board configuration with a specific BSP library, be sure to save it in the `config` folder of the BSP for which it was created. For example, `<install-dir>/bsp/pic32mz_ef_sk/config/`, as shown in the previous example, and be sure to select **Add to MPLAB Harmony Board List** so that the MHC will display it as a Custom Board Configuration for the associated BSP library. Give it a name that is descriptive of how it configures the board, so that you can easily understand why you might choose it over the default (or other) initial board configuration. Once saved, close and reopen the MHC and the new initial board Configuration Name will appear in the list of Custom Board Configurations for the associated BSP, as shown in the following image.



Board Support Packages

Provides information on the individual board support packages included in your installation of MPLAB Harmony.

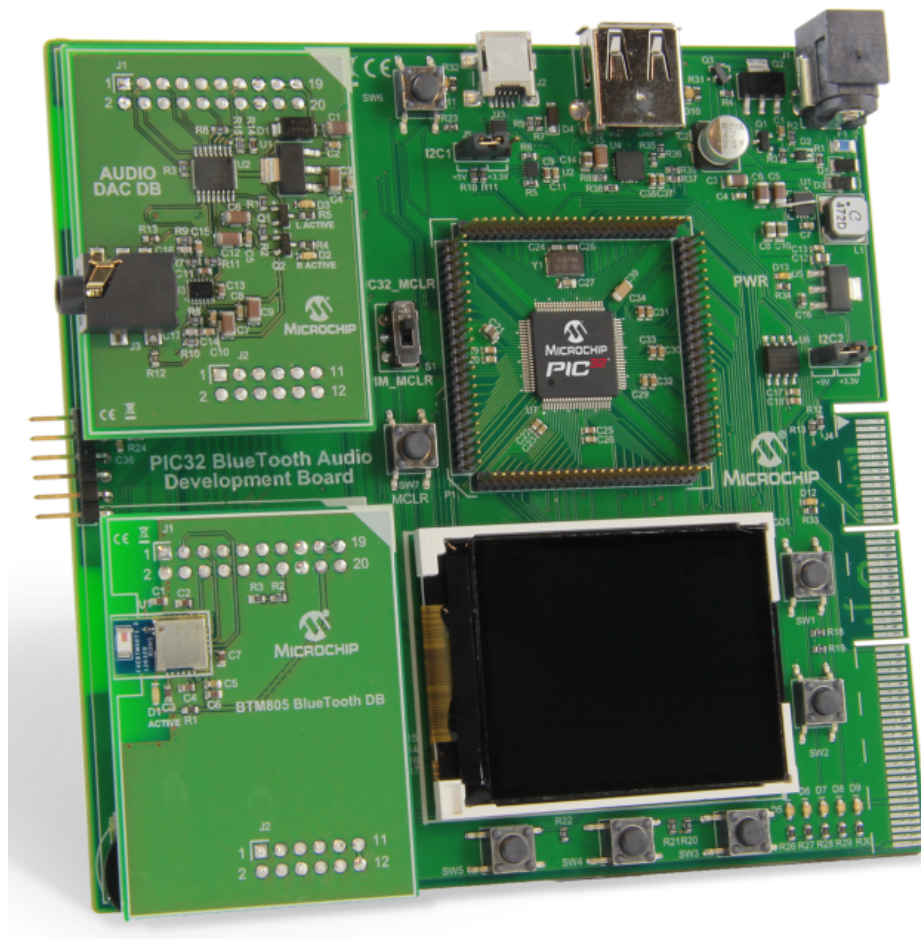
bt_audio_dk

PIC32 Bluetooth Audio Development Kit BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#).

The following figure illustrates the hardware configuration.

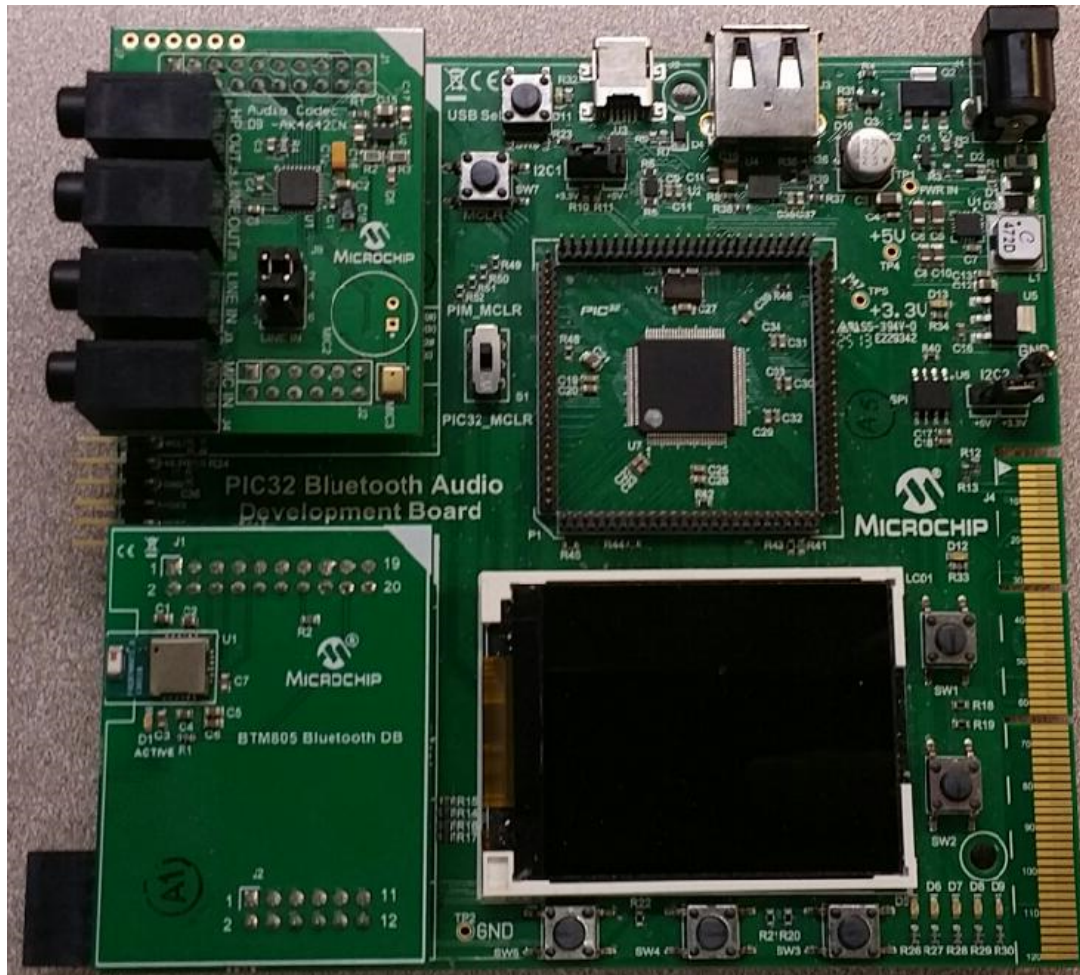


bt_audio_dk+4642

PIC32 Bluetooth Audio Development Kit with the AK4642 Audio Codec BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#) with the AK4642 Audio Codec. The following figure illustrates the hardware configuration.



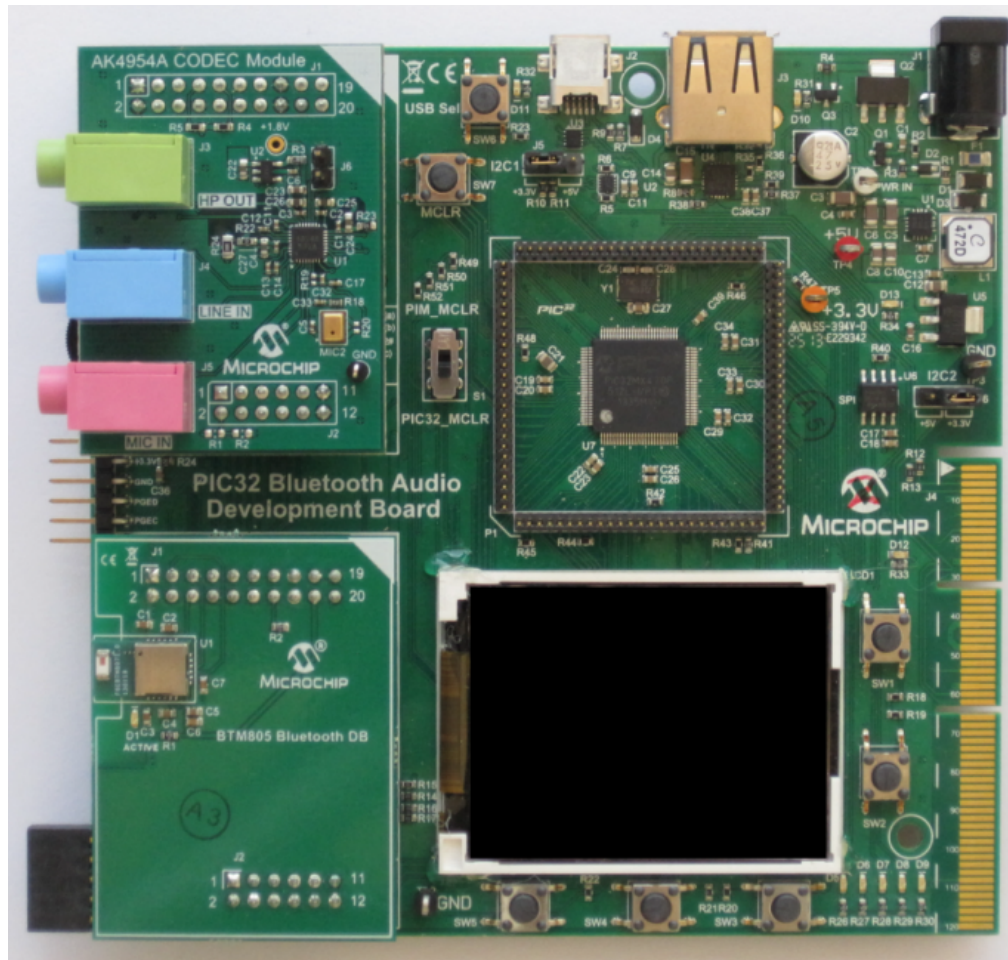
bt_audio_dk+ak4954

PIC32 Bluetooth Audio Development Kit with the AK4954 Audio Codec BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#) with the AK4954 Audio Codec.

The following figure illustrates the hardware configuration.



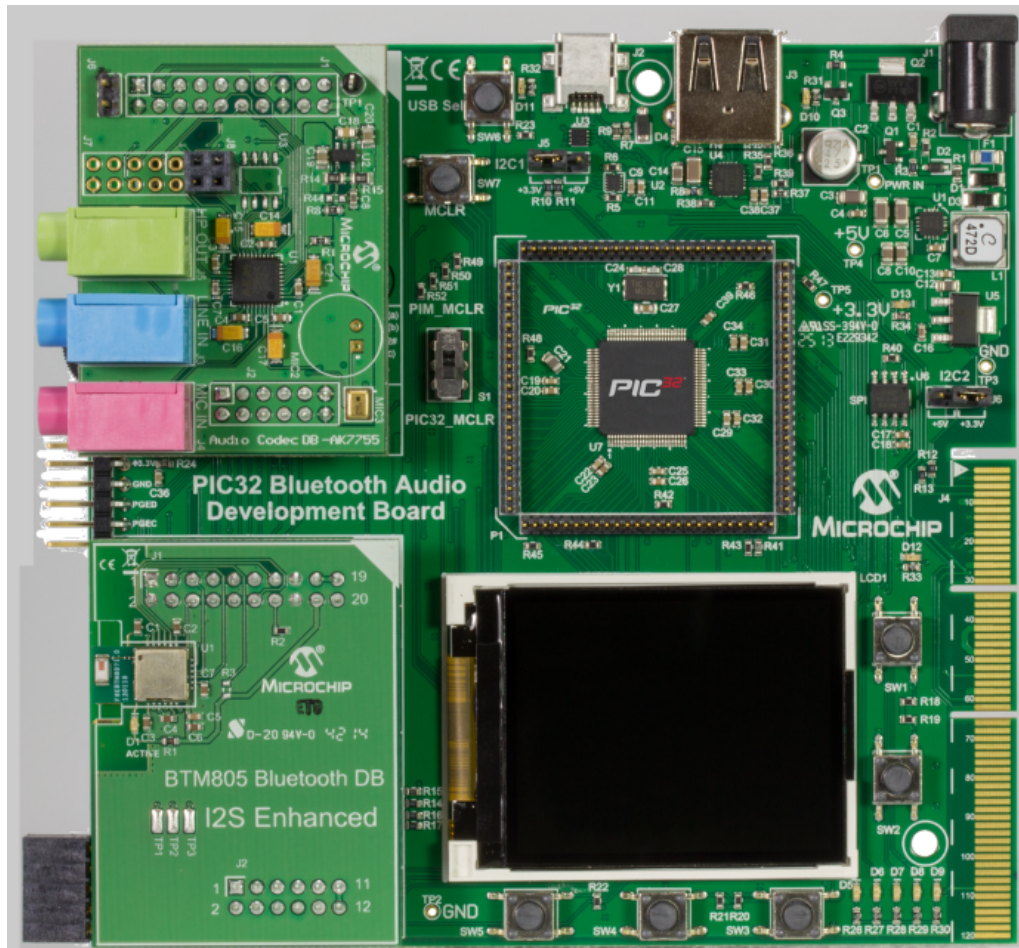
bt_audio_dk+ak7755

PIC32 Bluetooth Audio Development Kit with the AK7755 Audio Codec BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#) with the AK7755 Audio Codec.

The following figure illustrates the hardware configuration.



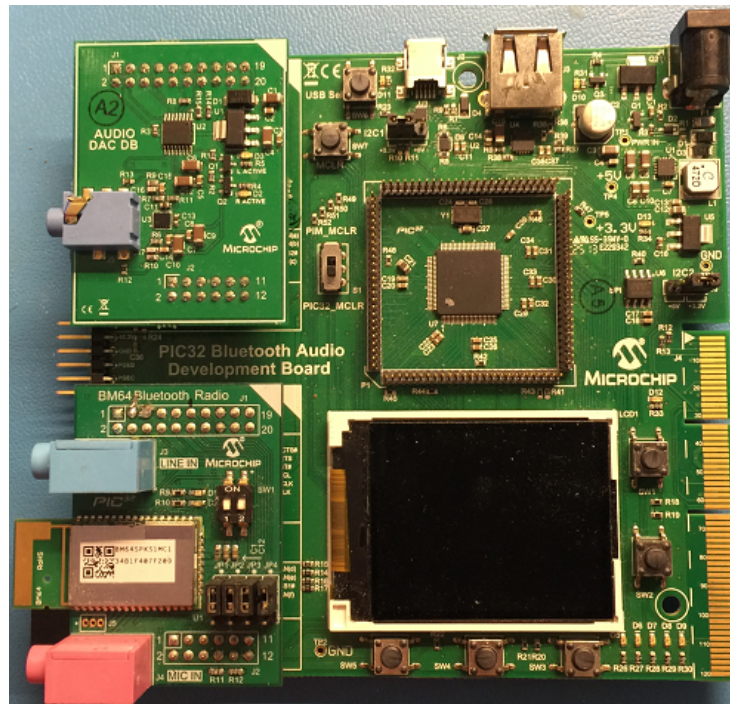
bt_audio_dk+bm64

PIC32 Bluetooth Audio Development Kit with the BM64 Bluetooth Module BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#) with the [BM64 Bluetooth Module Daughter Board](#) and the default [PIC32 Audio DAC Daughter Board](#).

The following figure illustrates the hardware configuration.



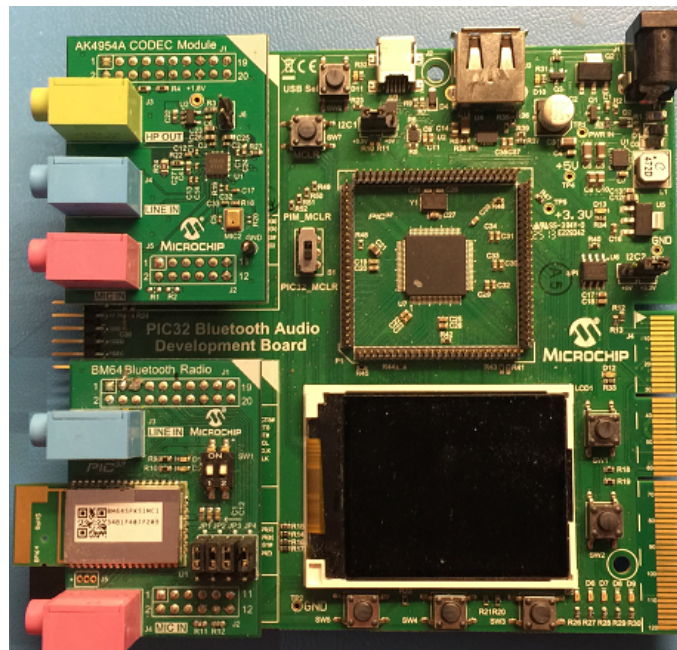
bt_audio_dk+bm64+ak4954

PIC32 Bluetooth Audio Development Kit with the BM64 Bluetooth Module and the AK4954 Audio Codec BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#) with the [BM64 Bluetooth Module Daughter Board](#) and the [PIC32 Audio Codec Daughter Board - AK4954A](#).

The following figure illustrates the hardware configuration.

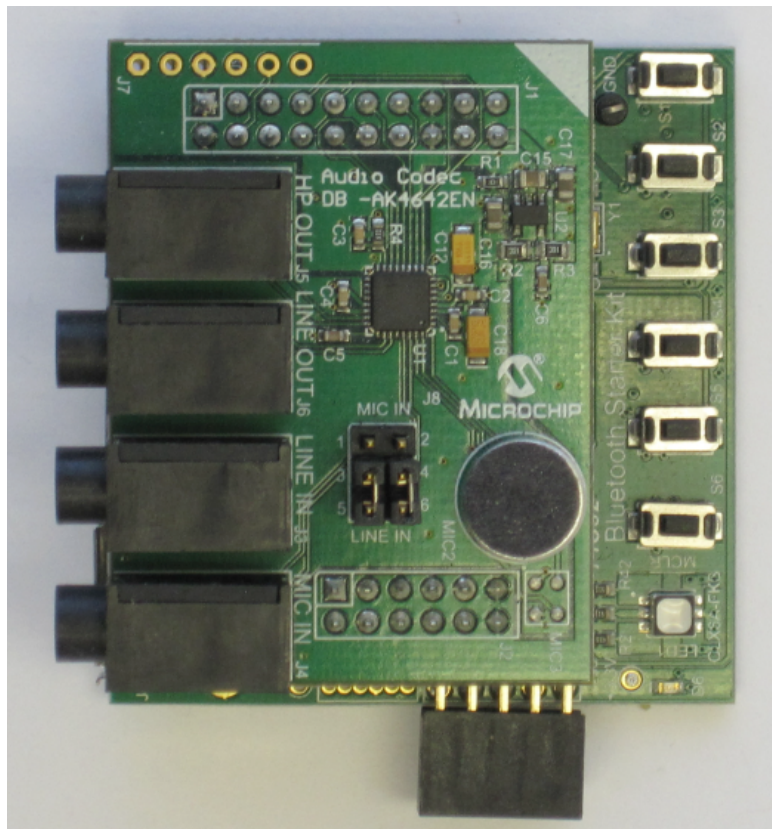


bt_sk+4642

PIC32 Bluetooth Starter Kit with the AK4642 Audio Codec BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#) with the AK4642 Audio Codec. The following figure illustrates the hardware configuration.

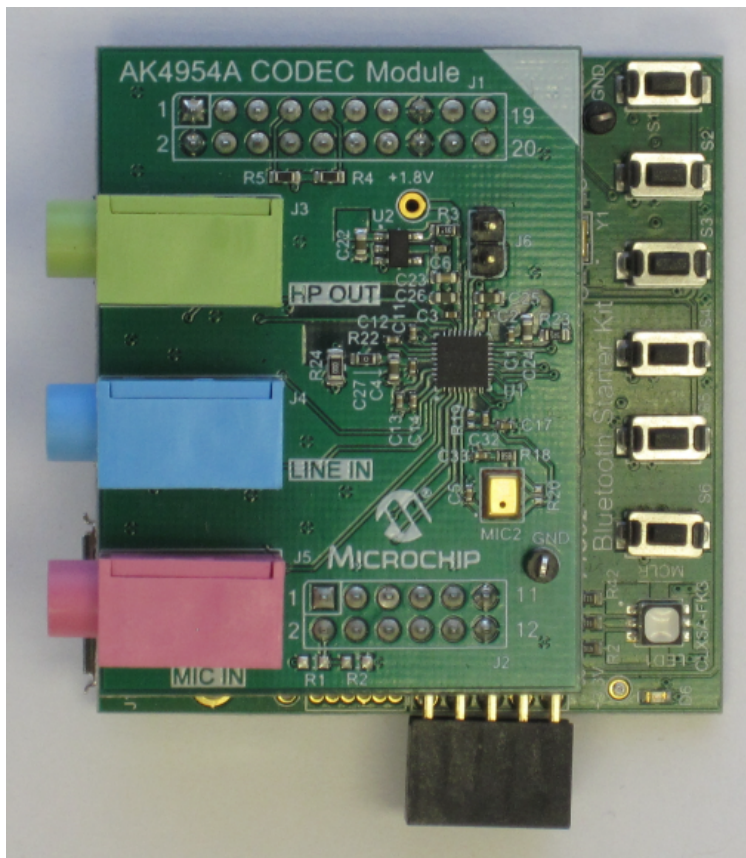


bt_sk+4954

PIC32 Bluetooth Starter Kit with the AK4954 Audio Codec BSP

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#) with the AK4954 Audio Codec. The following figure illustrates the hardware configuration.



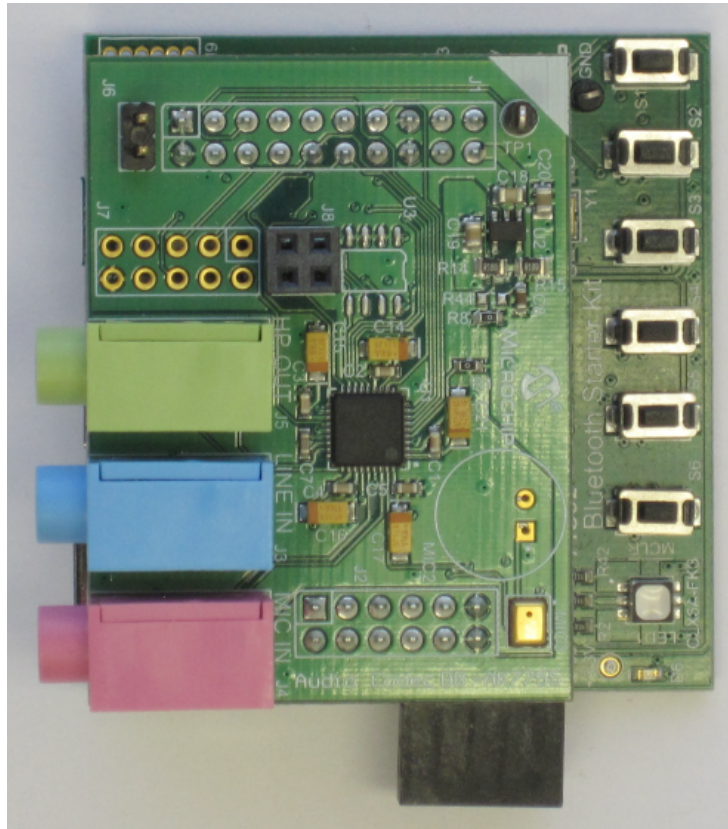
bt_sk+ak7755

PIC32 Bluetooth Starter Kit with the AK7755 Audio Codec BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Audio Development Kit](#) with the AK7755 Audio Codec.

The following figure illustrates the hardware configuration.



chipkit_wf32

chipKIT™ WF32™ Wi-Fi Development Board BSP.

Description

This BSP is intended for the [chipKIT™ WF32™ Wi-Fi Development Board](#).



Note:

If a USB Host application is used, the board *will not* be able to power the USB device without one of the following:

Using an external power supply (9V or greater) connected to J17

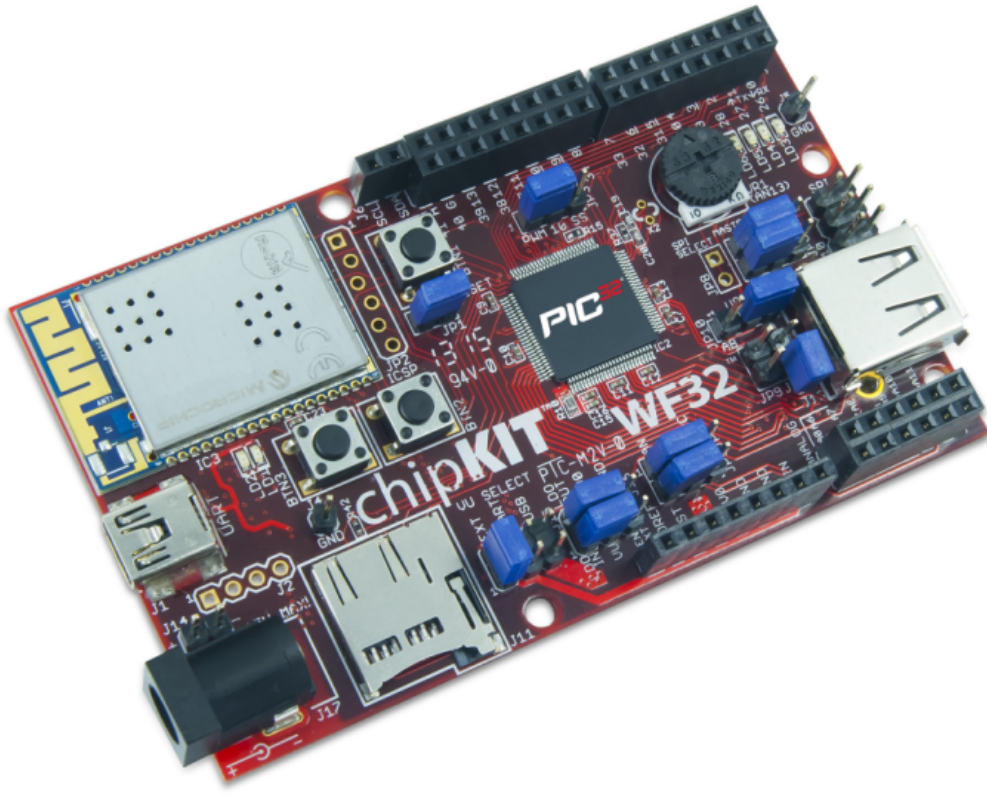
- or -

Bypassing the on-board voltage regulator by removing the jumpers on J16 and only connecting VU to 5V0.



Warning: Do not connect an external power supply in this configuration, or the 5V rail on the board will be supplied with the external voltage directly, which could result in damage to the board.

The following figure illustrates the hardware configuration.

**chipkit_wifire**

chipKIT™ Wi-FIRE Development Board BSP.

Description

This BSP is intended for the [chipKIT Wi-FIRE Development Board](#).



Note:

If a USB Host application is used, the board *will not* be able to power the USB device without one of the following:

Using an external power supply (9V or greater) connected to J15

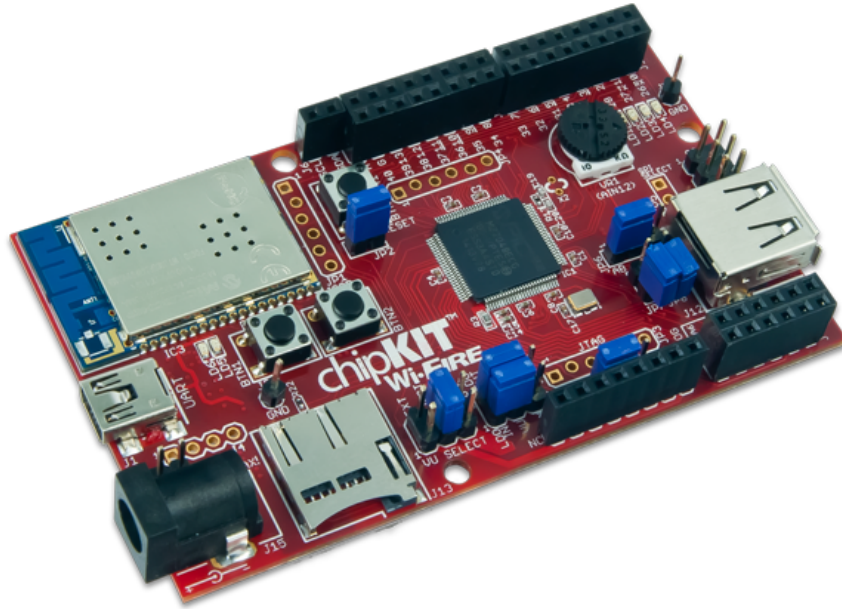
- or -

Bypassing the on-board voltage regulator by removing the jumpers on J17 and only connecting VU to 5V0.



Warning: Do not connect an external power supply in this configuration, or the 5V rail on the board will be supplied with the external voltage directly, which could result in damage to the board.

The following figure illustrates the hardware configuration.



pic32_gdb_ef

PIC32 Graphics Discovery Development Board BSP.

Description

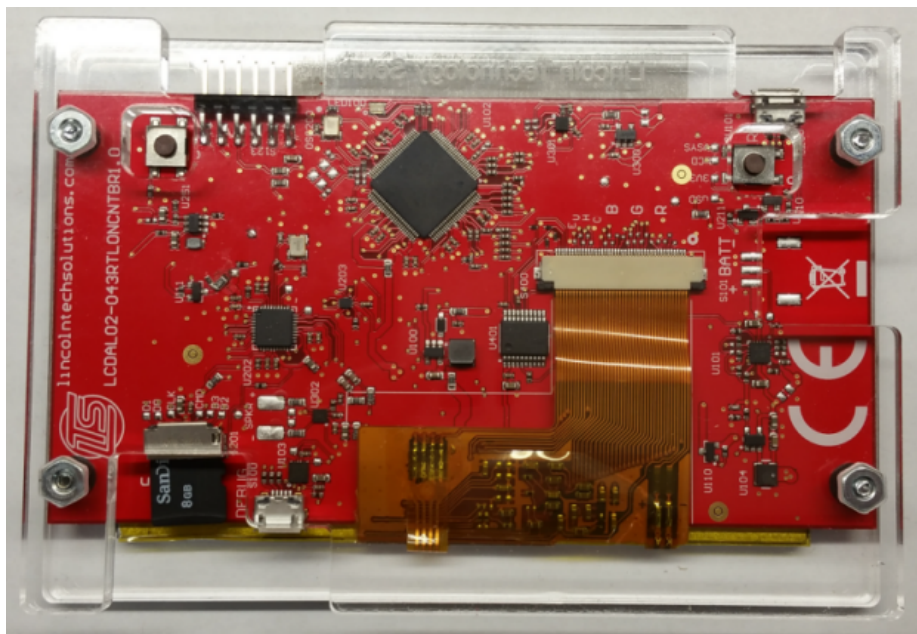
This BSP is intended for the the PIC32 Graphics Discovery Development Board and the [PIC32MZ EF Starter Kit](#).



Note:

Please contact your local Microchip sales office for information on obtaining this development board.

The following figure illustrates the hardware configuration.



pic32mk_gp_db

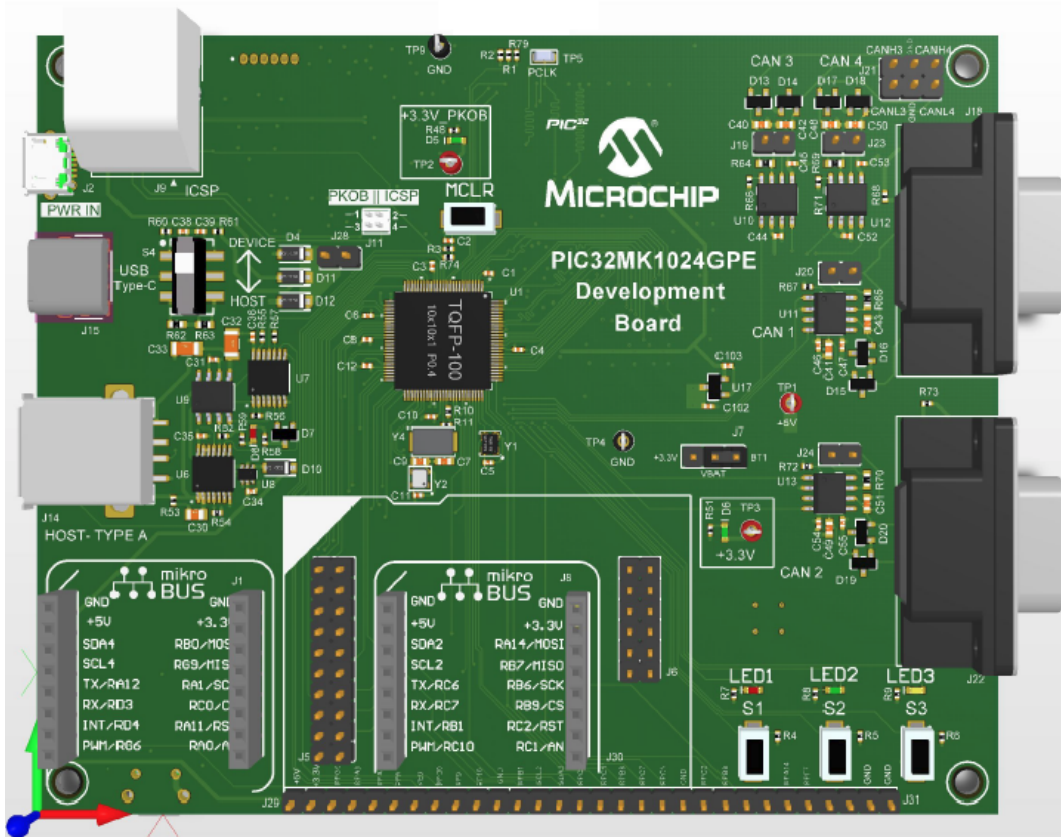
PIC32MK GP Development Board BSP

Description

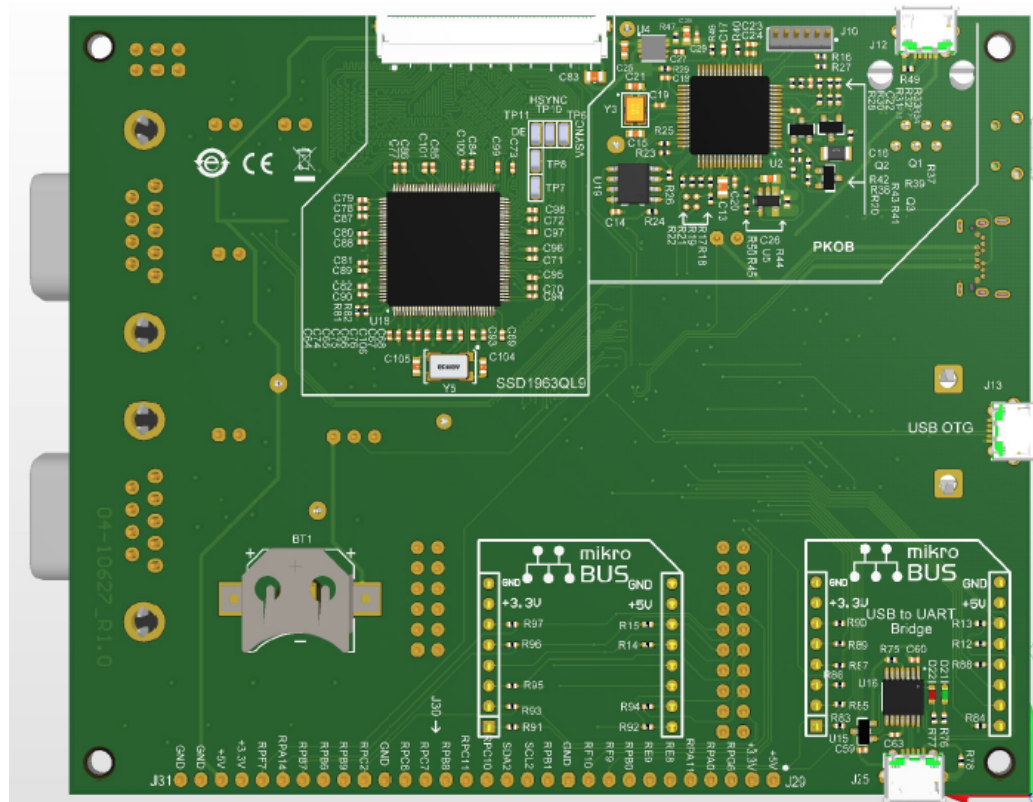
This BSP is intended for the [PIC32MK GP Development Board](#).

The following figures illustrate the hardware configuration.

Top View



Bottom View

***pic32mk_gp_db+wqvga_mxt***

PIC32MK GP Development Board (with SSD1963 Graphics Controller) and a 480x272 WQVGA maXTouch Display Module BSP.

Description

This BSP is intended for the [PIC32MK GP Development Board](#) (with SSD1963 Graphics Controller) connected to a [High-Performance 4.3" WQVGA Display Module with maXTouch](#).



Note: The display plus an interposer for use with the Multimedia Expansion Board II (MEBII) can be ordered from Microchip using the part number: AC320005-4.

The following figure illustrates the hardware configuration.



pic32mk_gp_db+vvga_mxt

PIC32MK GP Development Board (with SSD1963 Graphics Controller) and a [High-Performance WVGA Display Module with maXTouch](#).

Description

This BSP is intended for the [PIC32MK GP Development Board](#) (with SSD1963 Graphics Controller) connected to a [High-Performance WVGA Display Module with maXTouch](#).



Note: The display plus an interposer for use with the Multimedia Expansion Board II (MEB II) can be ordered from Microchip using the part number: AC320005-5.

The following figure illustrates the hardware configuration.



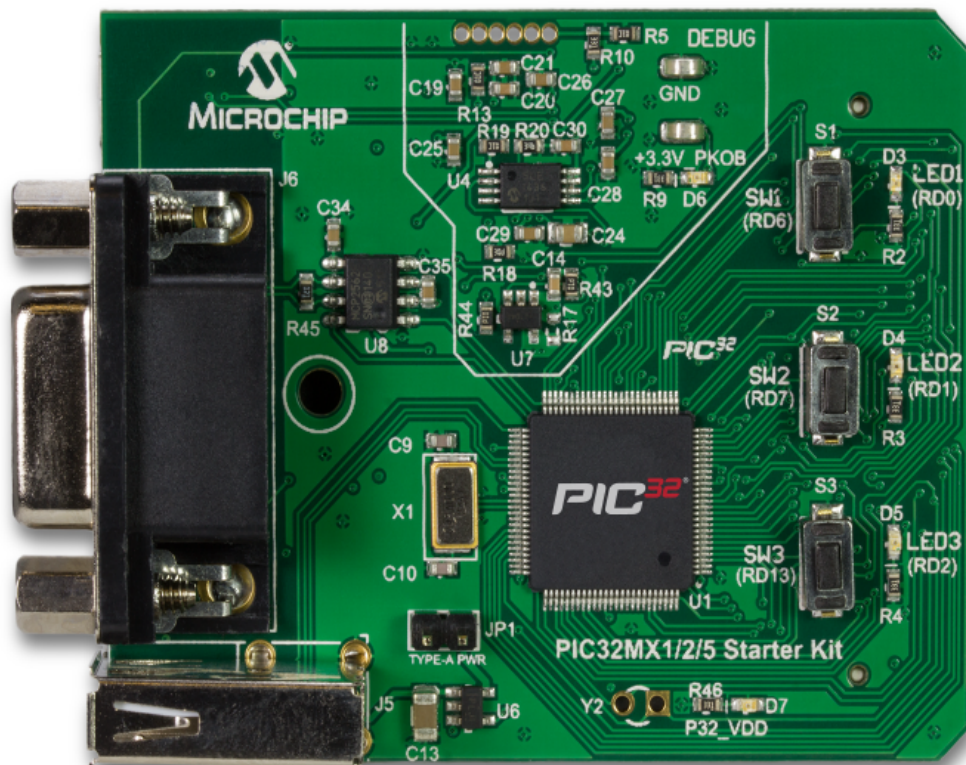
pic32mx_125_sk

PIC32MX1/2/5 Starter Kit BSP.

Description

This BSP is intended for the [PIC32MX1/2/5 Starter Kit](#).

The following figure illustrates the hardware configuration.



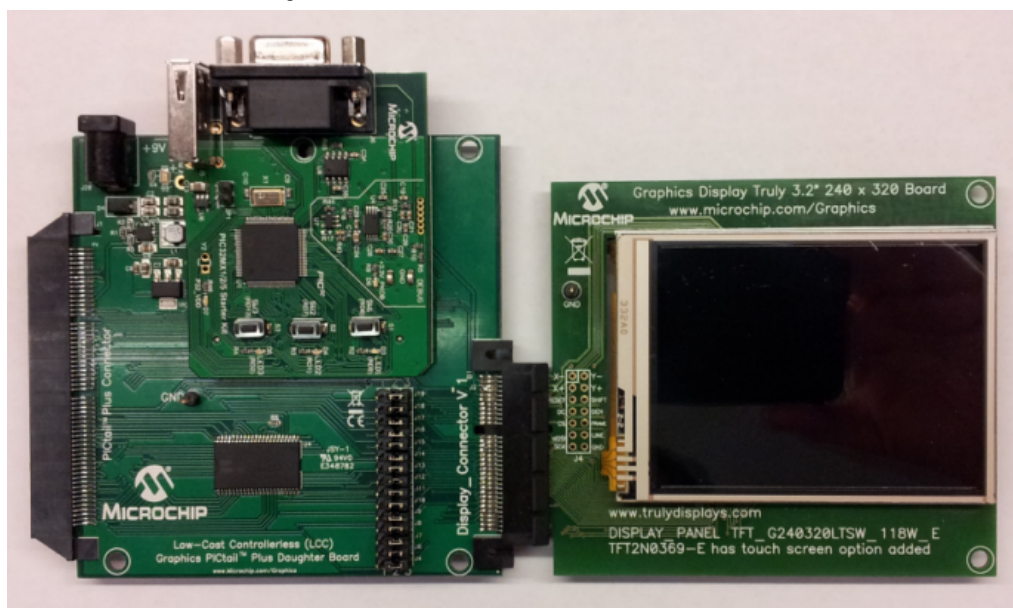
pic32mx_125_sk+lcc_pictail+qvga

PIC32MX1/2/5 Starter Kit plus the Low-Cost Controllerless (LCC) Graphics PICTail Plus Daughter Board with the Graphics Display Truly 3.2" 320x240 Board BSP.

Description

This BSP is intended for the [Low-Cost Controllerless \(LCC\) Graphics PICTail Plus Daughter Board](#) with the [Graphics Display Truly 3.2" 320x240 Board](#) connected to the [PIC32MX1/2/5 Starter Kit](#).

The following figure illustrates the hardware configuration.



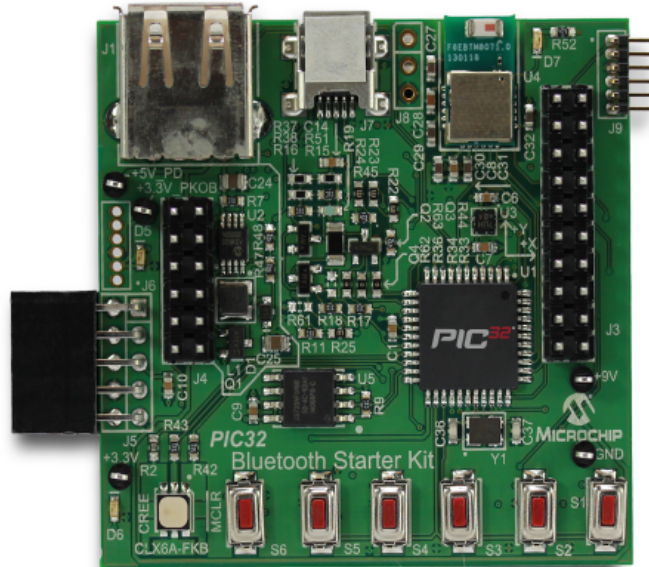
pic32mx_bt_sk

PIC32 Bluetooth Starter Kit BSP.

Description

This BSP is intended for the [PIC32 Bluetooth Starter Kit](#).

The following figure illustrates the hardware configuration.



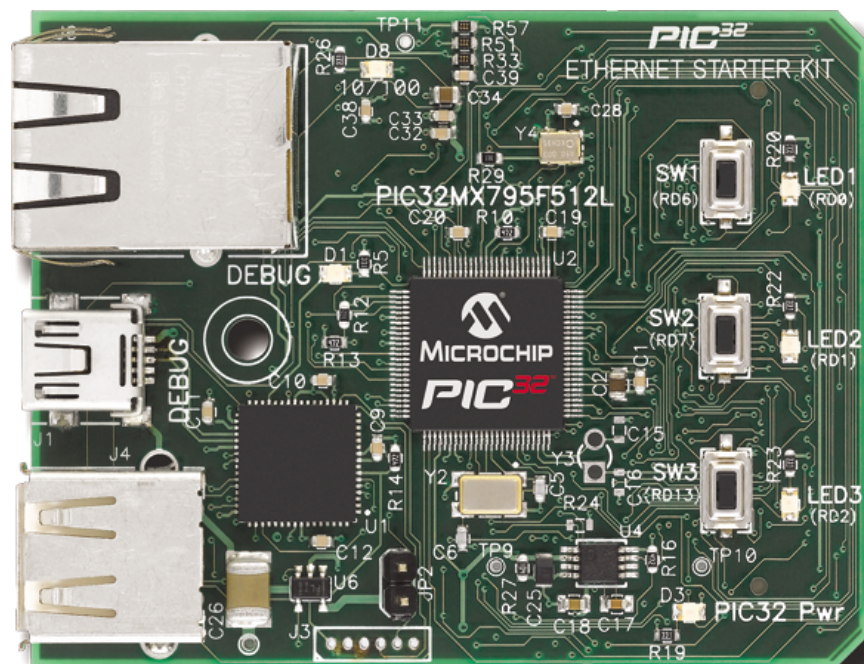
pic32mx_eth_sk

PIC32 Ethernet Starter Kit BSP.

Description

This BSP is intended for the [PIC32 Ethernet Starter Kit](#).

The following figure illustrates the hardware configuration.



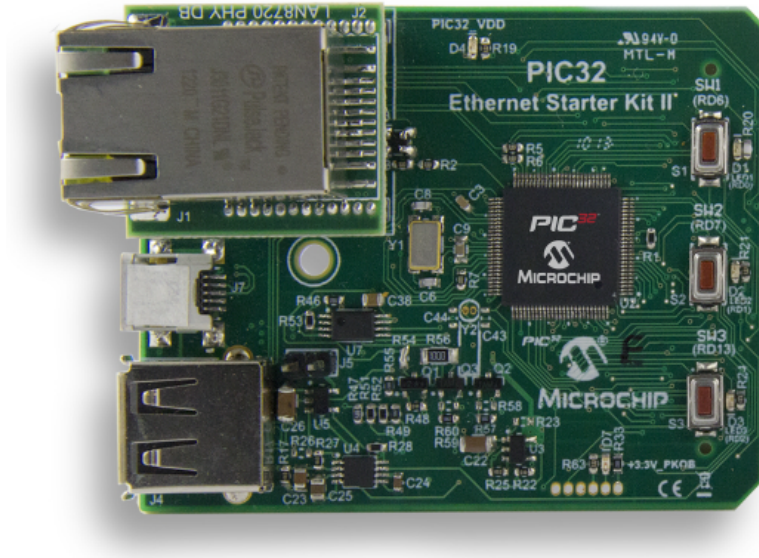
pic32mx_eth_sk2

PIC32 Ethernet Starter Kit II BSP.

Description

This BSP is intended for the [PIC32 Ethernet Starter Kit II](#).

The following figure illustrates the hardware configuration.



pic32mx_pcap_db

PIC32 GUI Development Board with Projected Capacitive Touch BSP.

Description

This BSP is intended for the [PIC32 GUI Development Board with Projected Capacitive Touch](#).

The following figure illustrates the hardware configuration.



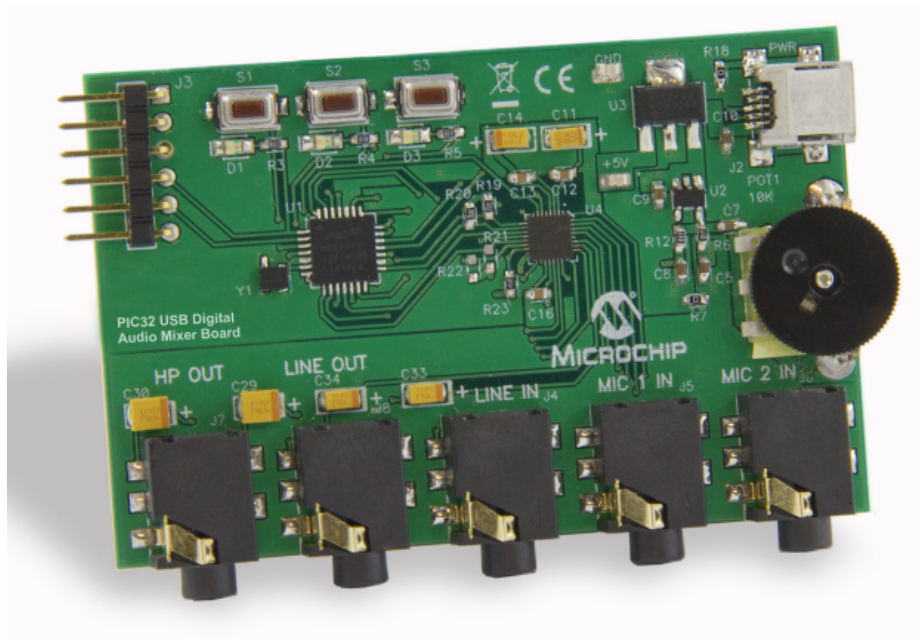
pic32mx_usb_digital_audio_ab

PIC32 USB Digital Audio Accessory Board BSP.

Description

This BSP is intended for the [PIC32 USB Digital Audio Accessory Board](#).

The following figure illustrates the hardware configuration.



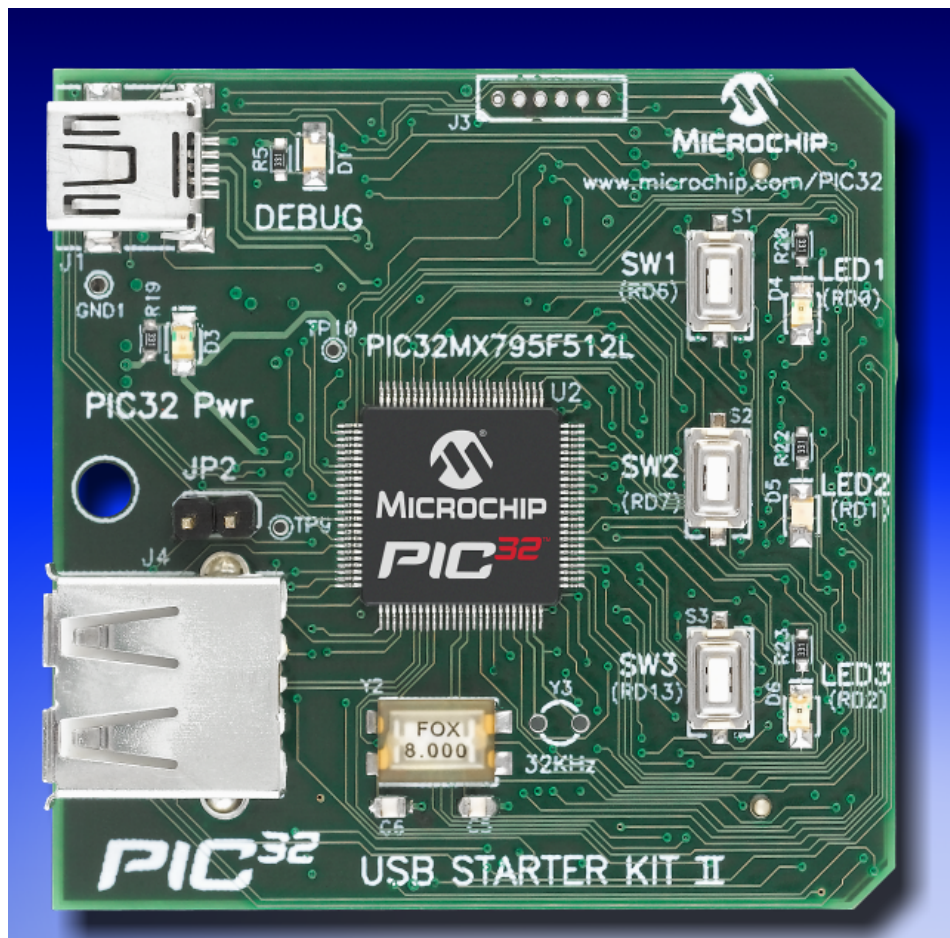
`pic32mx_usb_sk2`

PIC32 USB Starter Kit II BSP.

Description

This BSP is intended for the [PIC32 USB Starter Kit II](#).

The following figure illustrates the hardware configuration.



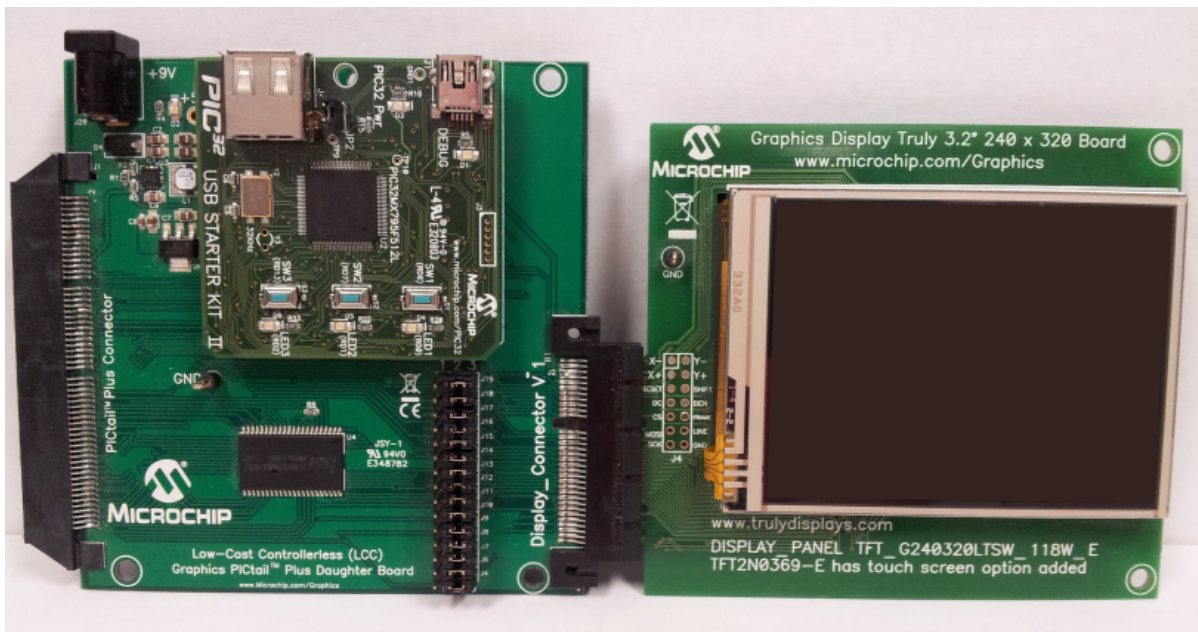
pic32mx_usb_sk2+lcc_pictail+qvga

PIC32 USB Starter Kit II plus the Low-Cost Controllerless (LCC) Graphics PICtail Plus Daughter Board with the Graphics Display Truly 3.2" 320x240 Board BSP.

Description

This BSP is intended for the [Low-Cost Controllerless \(LCC\) Graphics PICtail Plus Daughter Board](#) with the [Graphics Display Truly 3.2" 320x240 Board](#) connected to the [PIC32 USB Starter Kit II](#).

The following figure illustrates the hardware configuration.



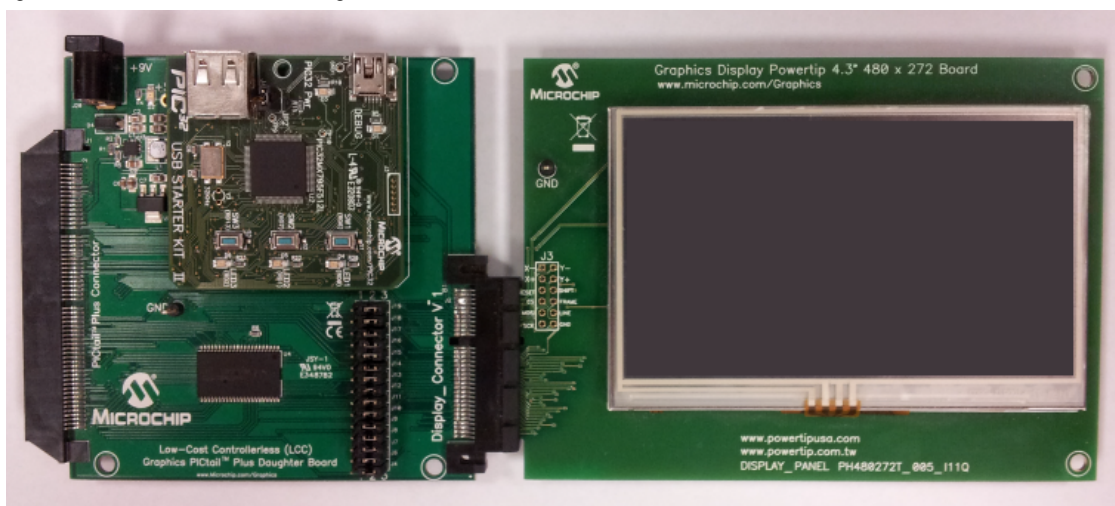
pic32mx_usb_sk2+lcc_pictail+wqvga

PIC32 USB Starter Kit II plus the Low-Cost Controllerless (LCC) Graphics PICtail Plus Daughter Board with Graphics Display Powertip 4.3" 480x272 Board BSP.

Description

This BSP is intended for the [Low-Cost Controllerless \(LCC\) Graphics PICtail Plus Daughter Board](#) with the [Graphics Display Powertip 4.3" 480x272 Board](#) connected to the [PIC32 USB Starter Kit II](#).

The following figure illustrates the hardware configuration.



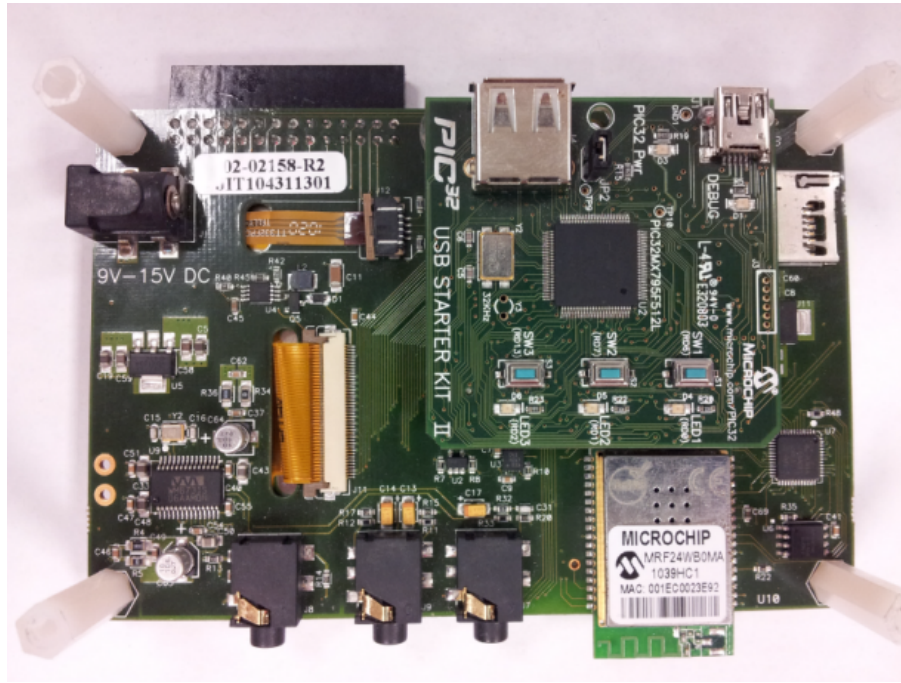
pic32mx_usb_sk2+meb

PIC32 USB Starter Kit II plus MEB BSP.

Description

This BSP is intended for the [Multimedia Expansion Board \(MEB\)](#) connected to the [PIC32 USB Starter Kit II](#).

The following figure illustrates the hardware configuration.



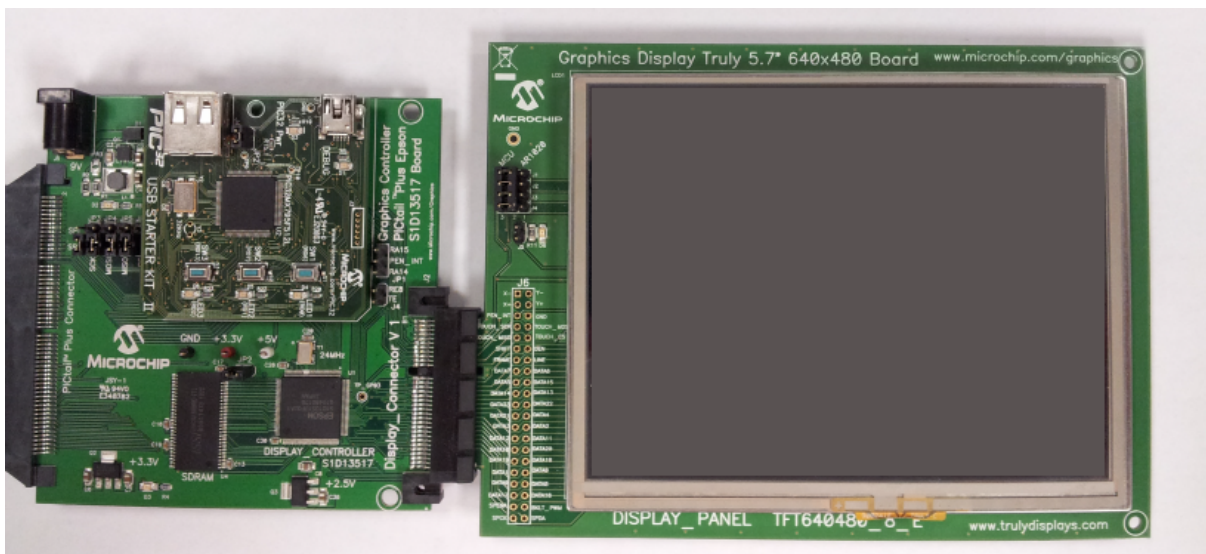
pic32mx_usb_sk2+s1d_pictail+vga

PIC32 USB Starter Kit II plus the Graphics Controller Pictail Plus Epson S1D13517 Daughter Board with Graphics Display Truly 5.7" 640x480 Board BSP.

Description

This BSP is intended for the [Graphics Controller Pictail Plus Epson S1D13517 Daughter Board](#) with the [Graphics Display Truly 5.7" 640x480 Board](#) connected to the [PIC32 USB Starter Kit II](#).

The following figure illustrates the hardware configuration.



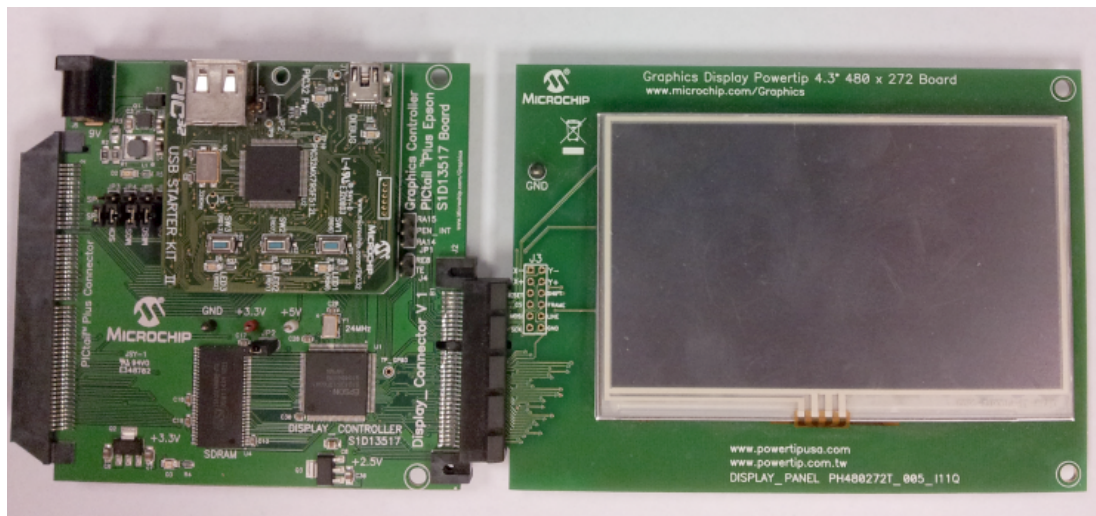
pic32mx_usb_sk2+s1d_pictail+wqvga

PIC32 USB Starter Kit II plus the Graphics Controller PICtail Plus Epson S1D13517 Daughter Board with the Graphics Display Powertip 4.3" 480x272 Board BSP.

Description

This BSP is intended for the [Graphics Controller PICtail Plus Epson S1D13517 Daughter Board](#) with the [Graphics Display Powertip 4.3" 480x272 Board](#) connected to the [PIC32 USB Starter Kit II](#).

The following figure illustrates the hardware configuration.



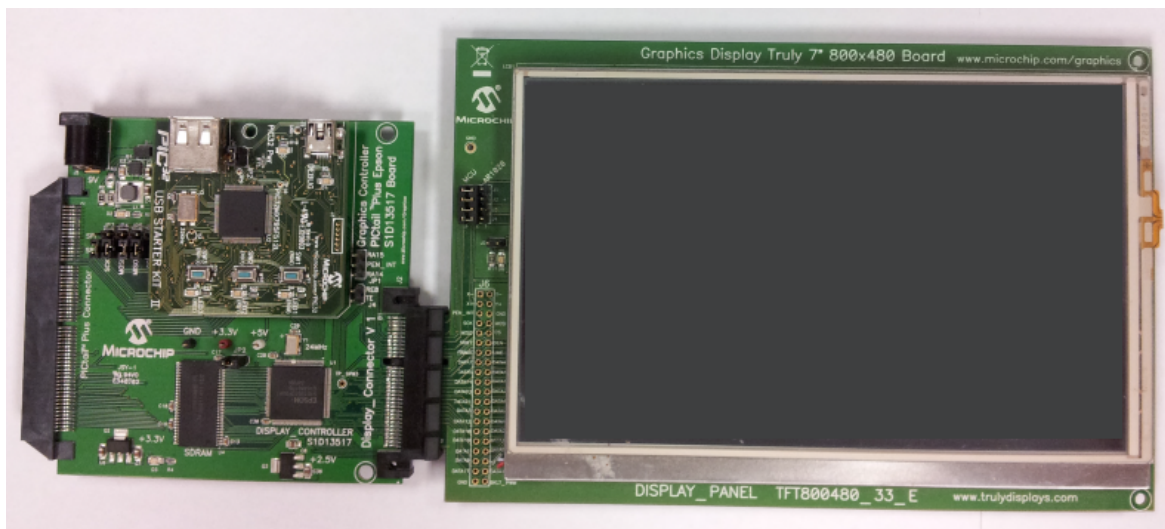
pic32mx_usb_sk2+s1d_pictail+vwga

PIC32 USB Starter Kit II plus the Graphics Controller PICtail Plus Epson S1D13517 Daughter Board with Graphics Display Truly 7" 800x480 Board BSP.

Description

This BSP is intended for the [Graphics Controller PICtail Plus Epson S1D13517 Daughter Board](#) with [Graphics Display Truly 7" 800x480 Board](#) connected to the [PIC32 USB Starter Kit II](#).

The following figure illustrates the hardware configuration.



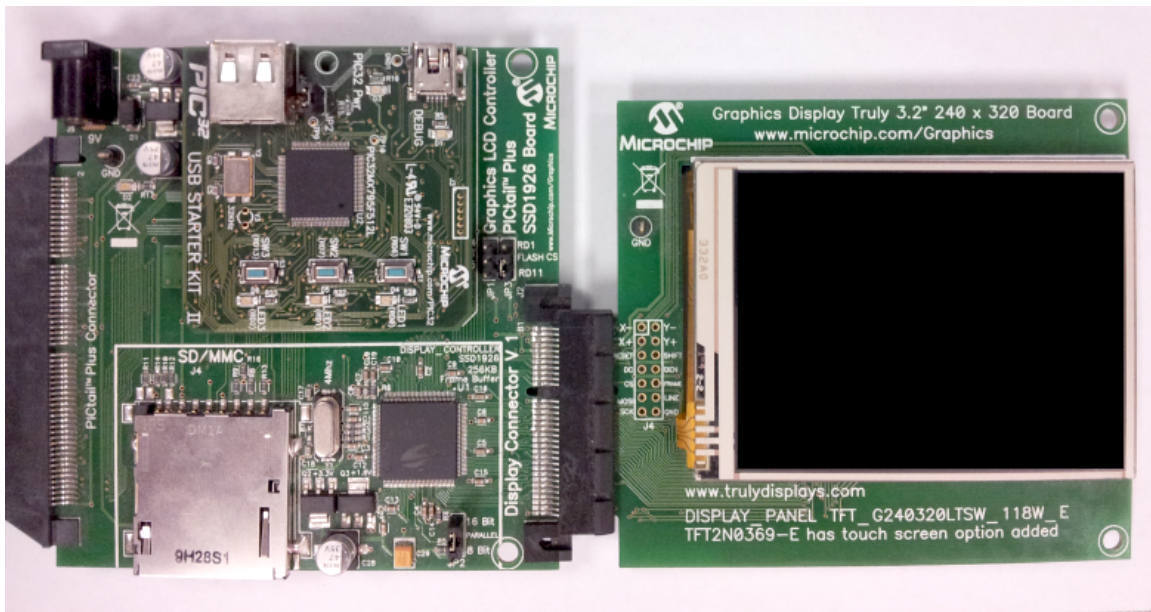
pic32mx_usb_sk2+ssd_pictail+qvga

PIC32 USB Starter Kit II plus the Graphics LCD Controller PICtail Plus SSD1926 Daughter Board with Graphics Display Truly 3.2" 320x240 Board BSP.

Description

This BSP is intended for the [Graphics LCD Controller PICtail Plus SSD1926 Daughter Board](#) with [Graphics Display Truly 3.2" 320x240 Board](#) connected to the [PIC32 USB Starter Kit II](#).

The following figure illustrates the hardware configuration.



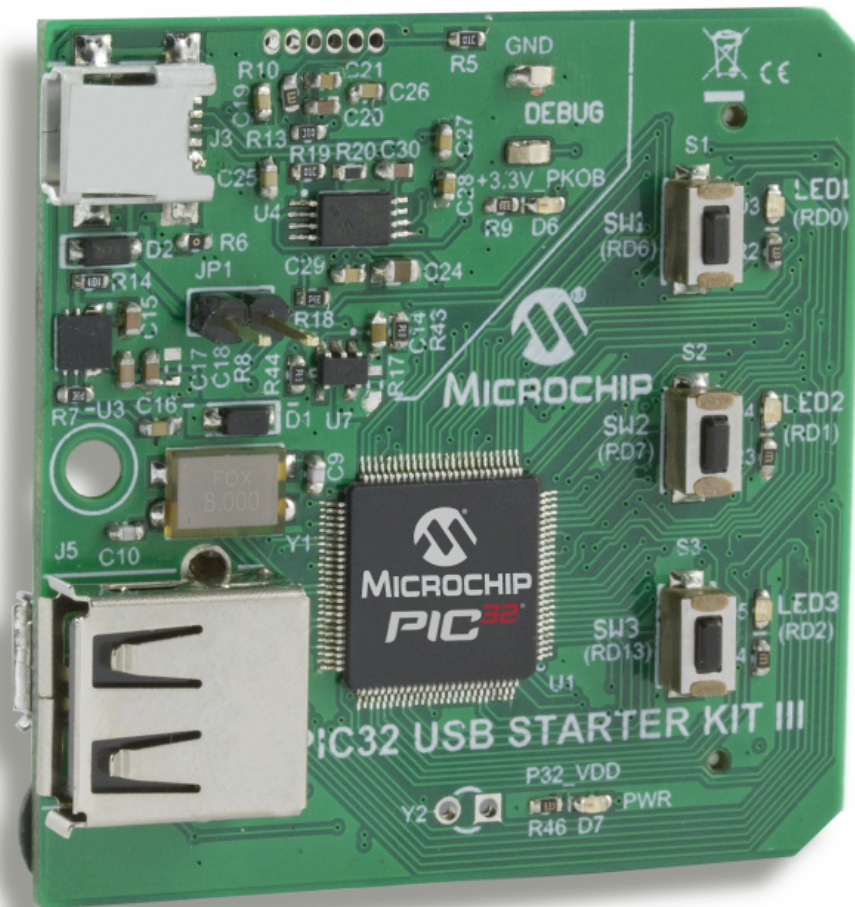
pic32mx_usb_sk3

PIC32 USB Starter Kit III BSP.

Description

This BSP is intended for the [PIC32 USB Starter Kit III](#).

The following figure illustrates the hardware configuration.



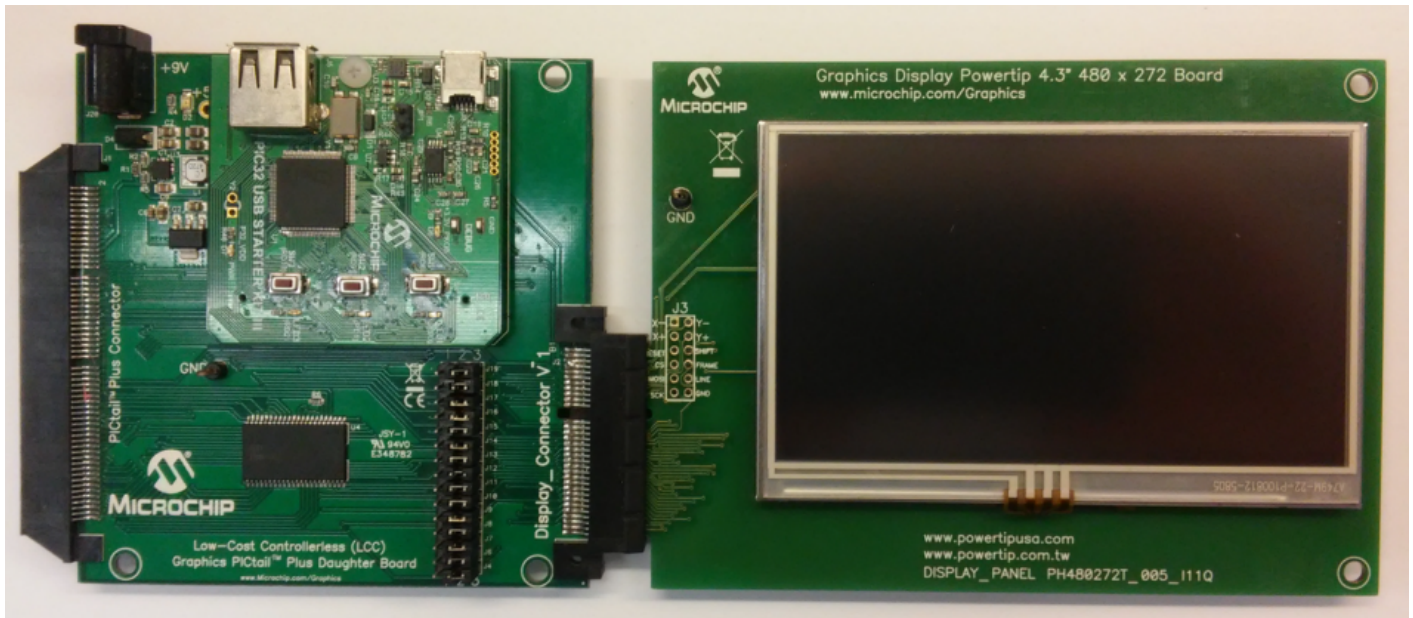
pic32mx_usb_sk3+lcc_pictail+wqvga

PIC32 USB Starter Kit III plus the Low-Cost Controllerless (LCC) Graphics PICtail Plus Daughter Board with Graphics Display Powertip 4.3" 480x272 Board BSP.

Description

This BSP is intended for the [Low-Cost Controllerless \(LCC\) Graphics PICtail Plus Daughter Board](#) with the [Graphics Display Powertip 4.3" 480x272 Board](#) connected to the [PIC32 USB Starter Kit III](#).

The following figures illustrates the hardware configuration.



pic32mx_xlp_sk

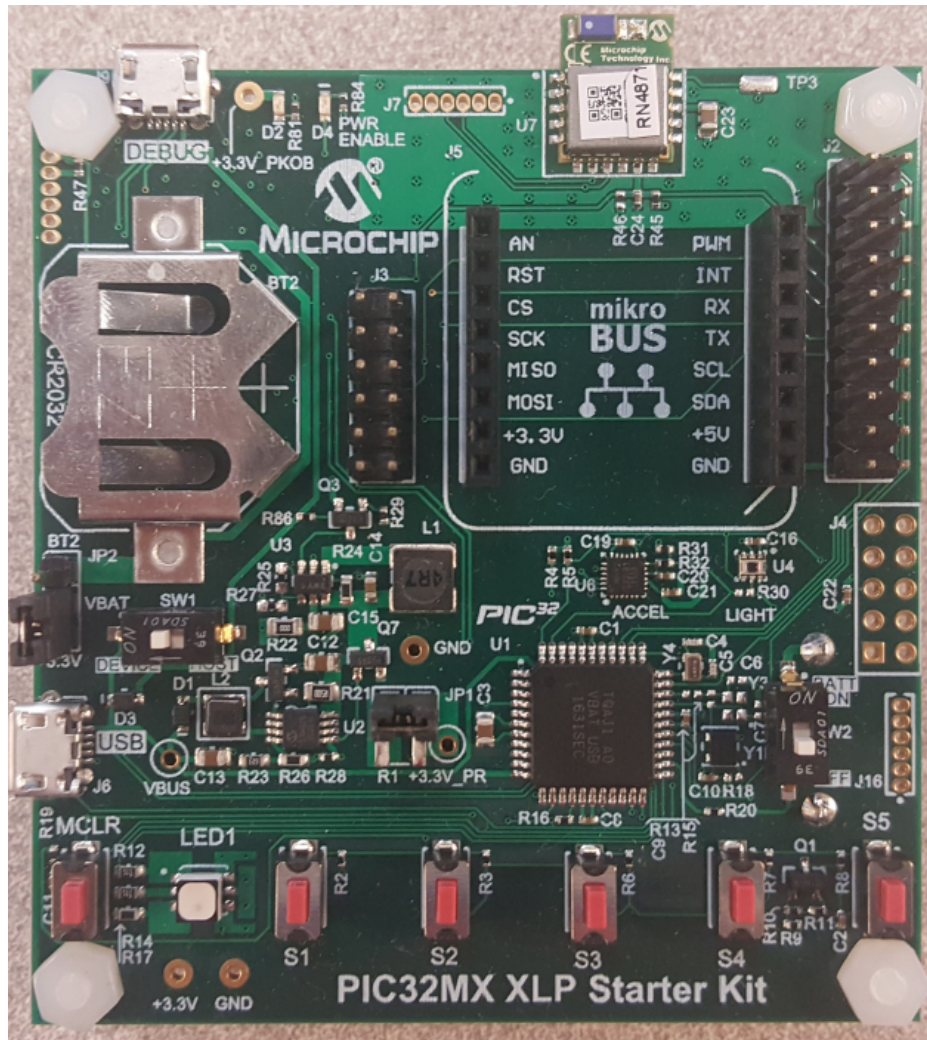
PIC32MX XLP Starter Kit BSP.

Description

This BSP is intended for the [PIC32MX XLP Starter Kit](#).

The following figures illustrates the hardware configuration.

Top View



Bottom View

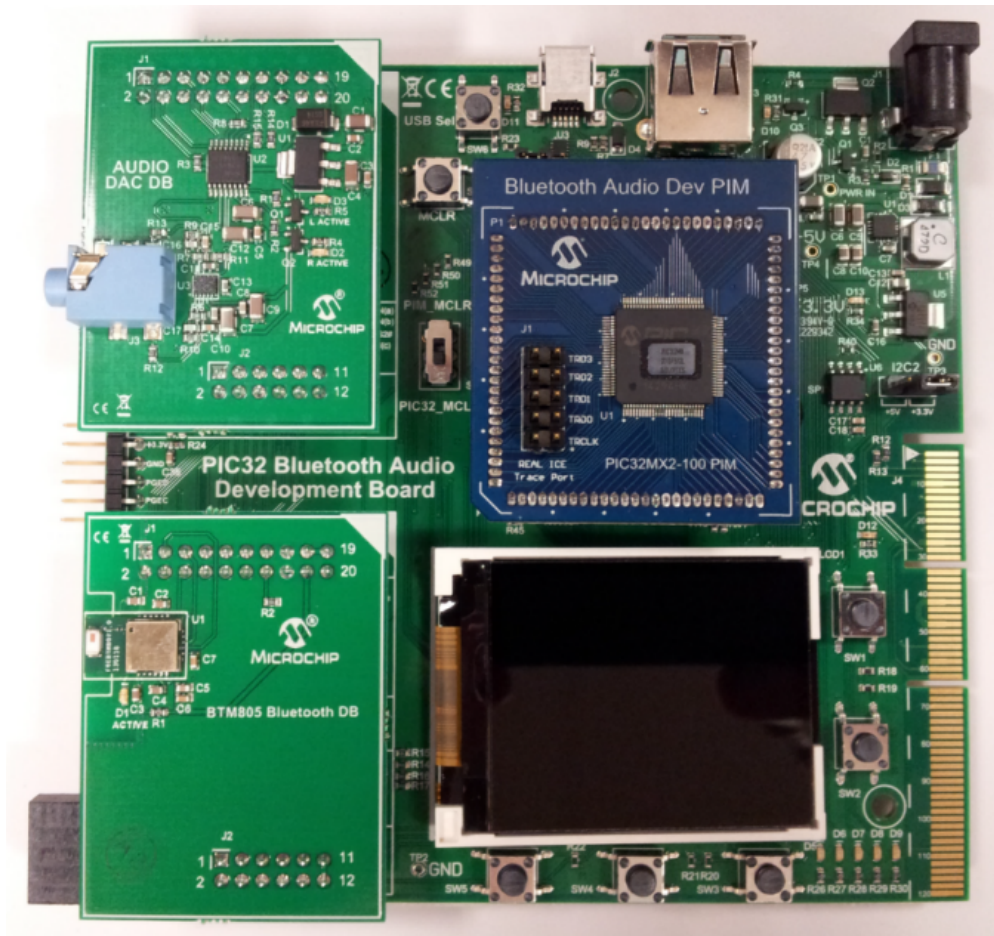


pic32mx270f512l_pim+bt_audio_dk

PIC32MX270F512L Plug-in Module (PIM) plus PIC32 Bluetooth Audio Development Kit.

Description

This BSP is intended for the [PIC32MX270F512L Plug-in Module \(PIM\)](#) connected to the [PIC32 Bluetooth Audio Development Kit](#). The following figure illustrates the hardware configuration.



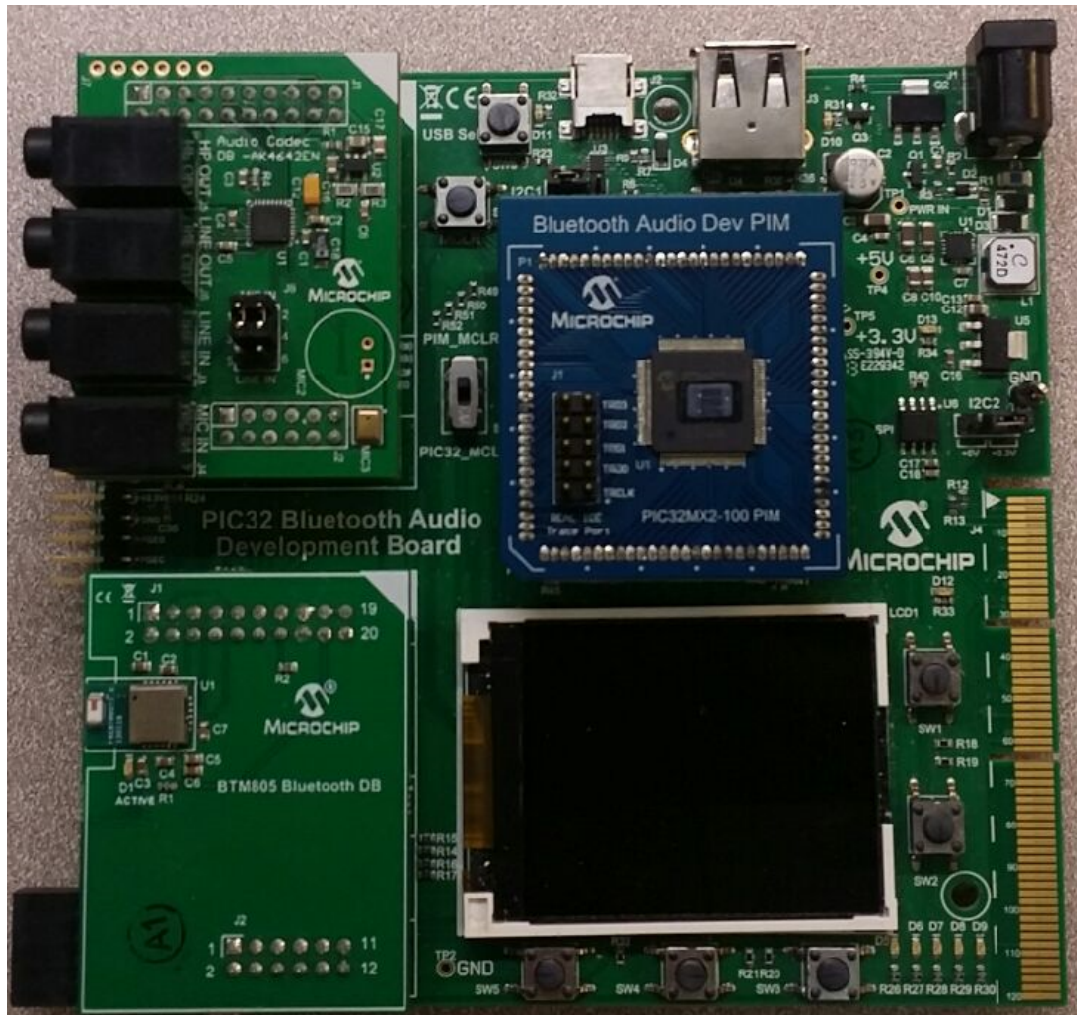
pic32mx_270f512l_pim+ bt_audio_dk+ak4642

PIC32MX270F512L Plug-in Module (PIM) plus PIC32 Bluetooth Audio Development Kit with the AK4642 Audio Codec.

Description

This BSP is intended for the [PIC32MX270F512L Plug-in Module \(PIM\)](#) connected to the [PIC32 Bluetooth Audio Development Kit](#) with the [Audio Codec Daughter Board AK4642EN](#).

The following figure illustrates the hardware configuration.

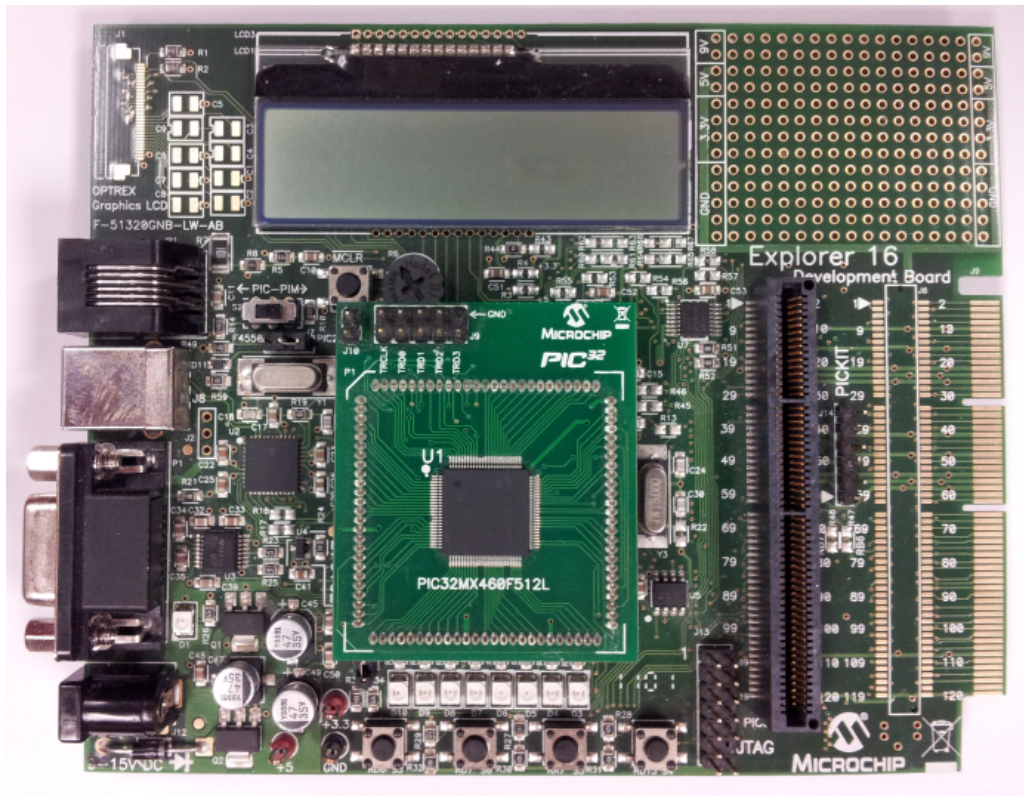


pic32mx460_pim+e16

PIC32MX460F512L Plug-in Module (PIM) plus Explorer 16 Development Board BSP.

Description

This BSP is intended for the [PIC32MX460F512L Plug-in Module \(PIM\)](#) connected to the [Explorer 16 Development Board](#). The following figure illustrates the hardware configuration.



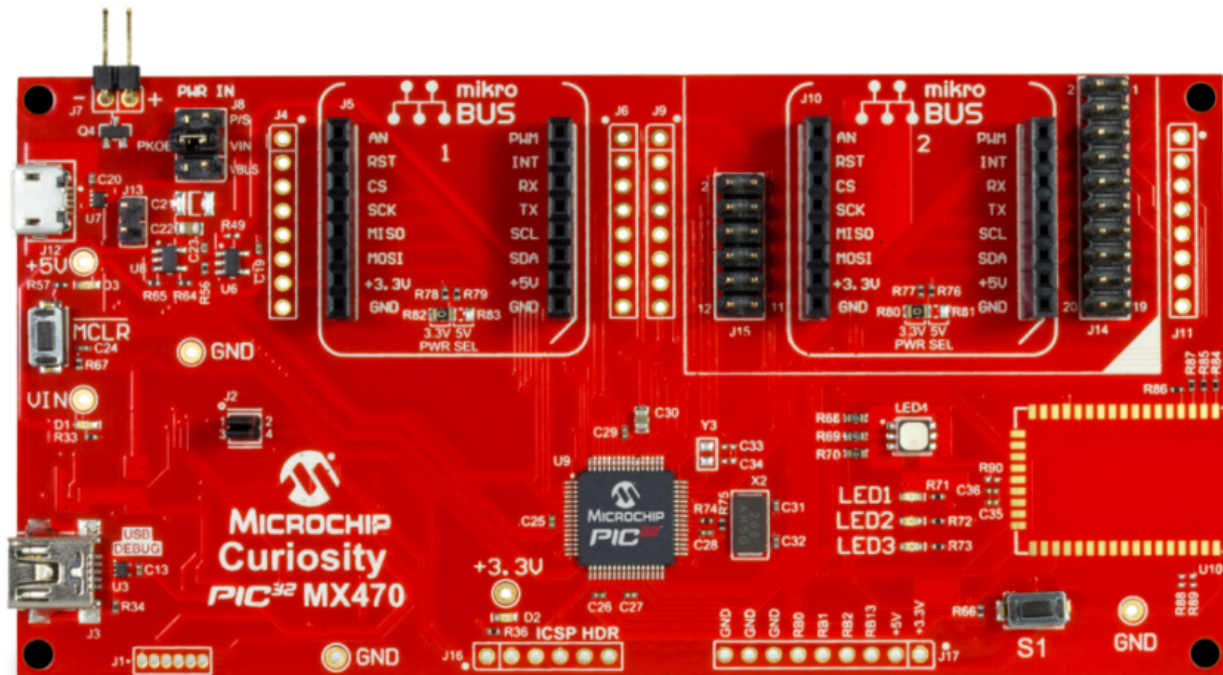
pic32mx470_curiosity

PIC32MX470 Curiosity Development Board BSP.

Description

This BSP is intended for the [PIC32MX470 Curiosity Development Board](#).

The following figure illustrates the hardware configuration.

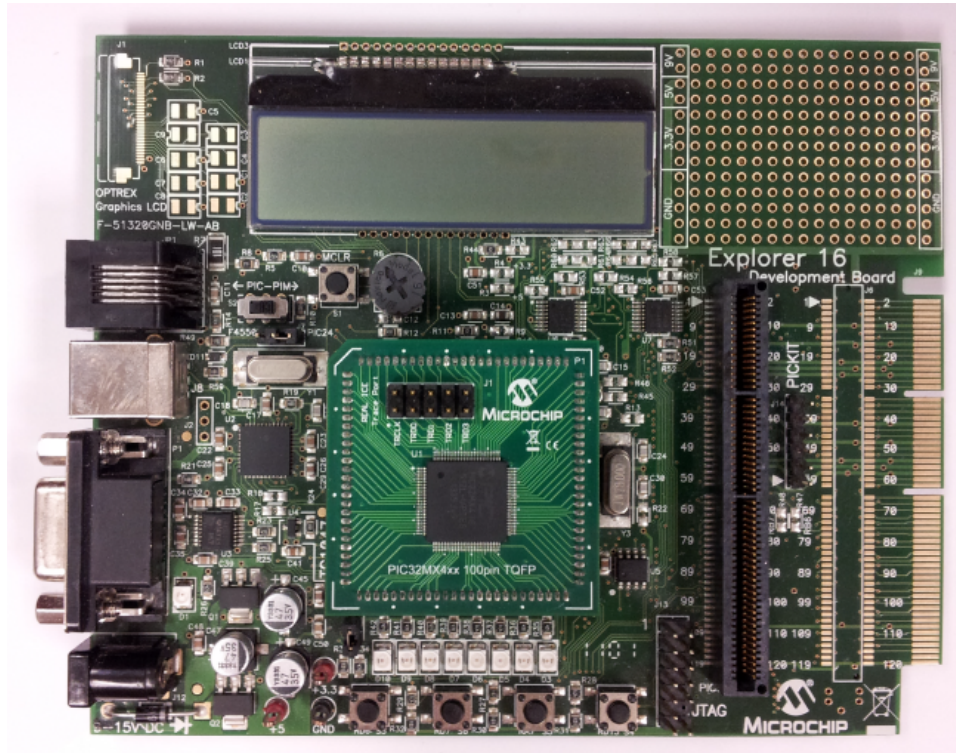


pic32mx470_pim+e16

PIC32MX450/470F512L Plug-in Module (PIM) plus Explorer 16 Development Board BSP.

Description

This BSP is intended for the [PIC32MX450/470F512L Plug-in Module \(PIM\)](#) connected to the [Explorer 16 Development Board](#). The following figure illustrates the hardware configuration.

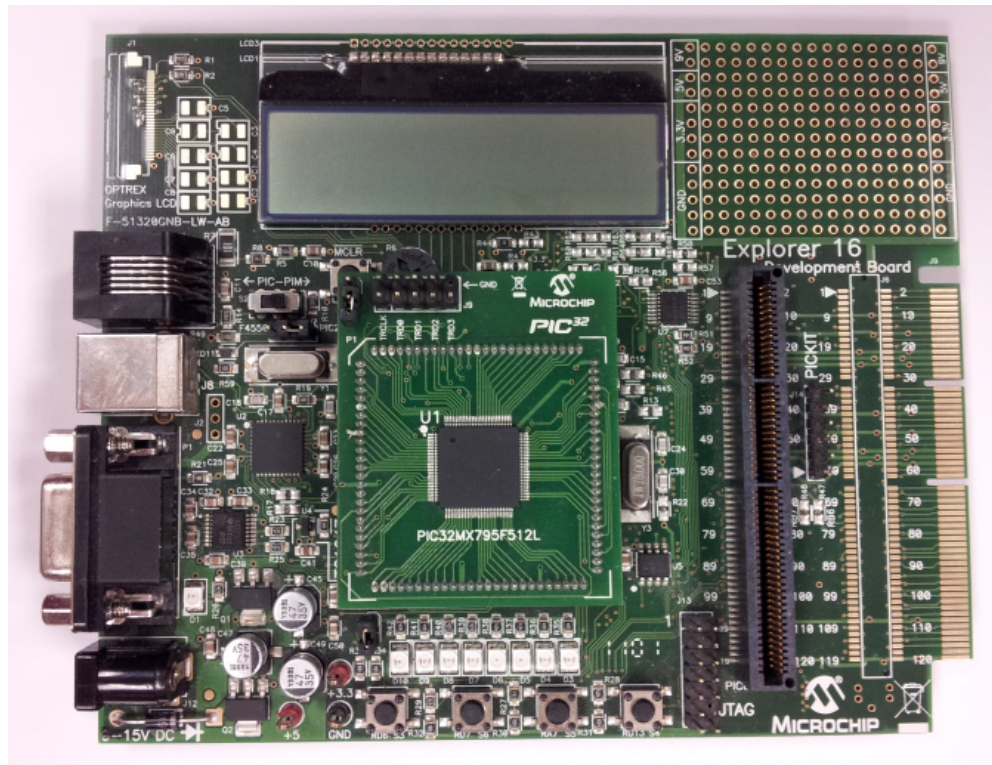


pic32mx795_pim+e16

PIC32MX795F512L Plug-in Module (PIM) plus Explorer 16 Development Board BSP.

Description

This BSP is intended for the [PIC32MX795F512L CAN-USB Plug-in Module \(PIM\)](#) connected to the [Explorer 16 Development Board](#). The following figure illustrates the hardware configuration.



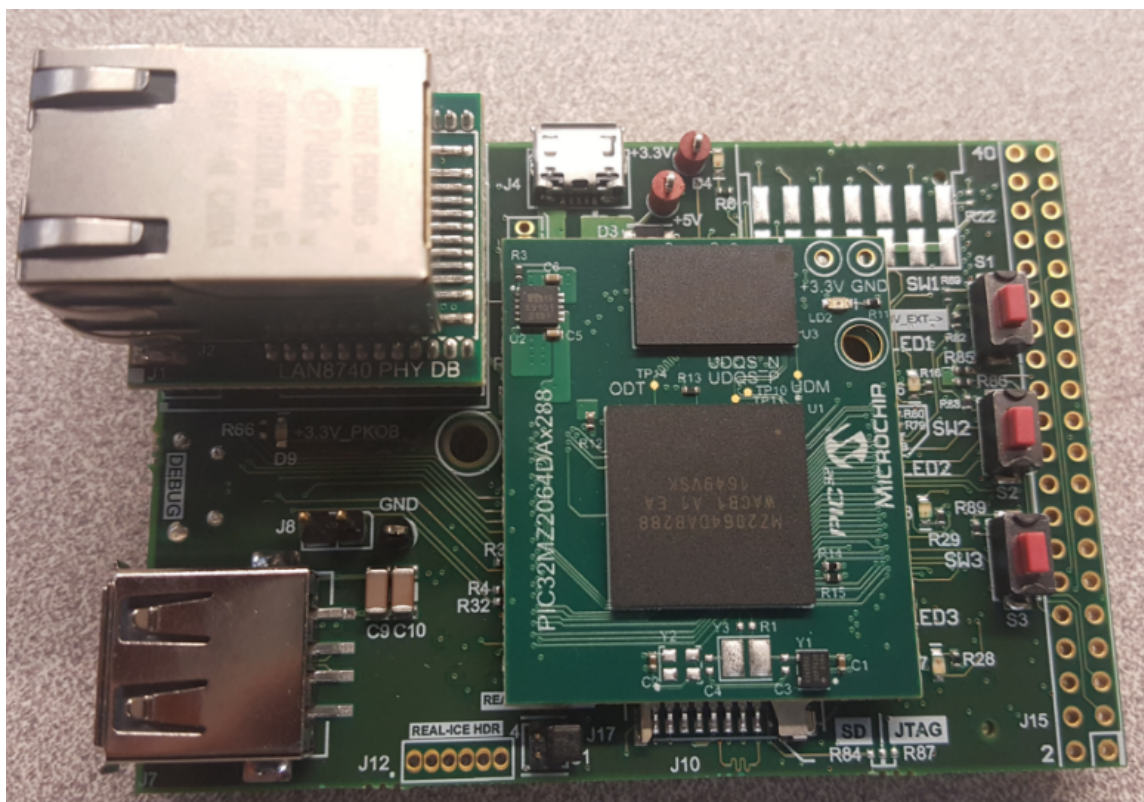
`pic32mz_da_sk_extddr`

PIC32MZ Embedded Graphics with External DRAM (DA) Starter Kit BSP.

Description

This BSP is intended for the [PIC32MZ Embedded Graphics with External DRAM \(DA\) Starter Kit](#).

The following figure illustrates the hardware configuration.



pic32mz_da_sk_extddr+meb2

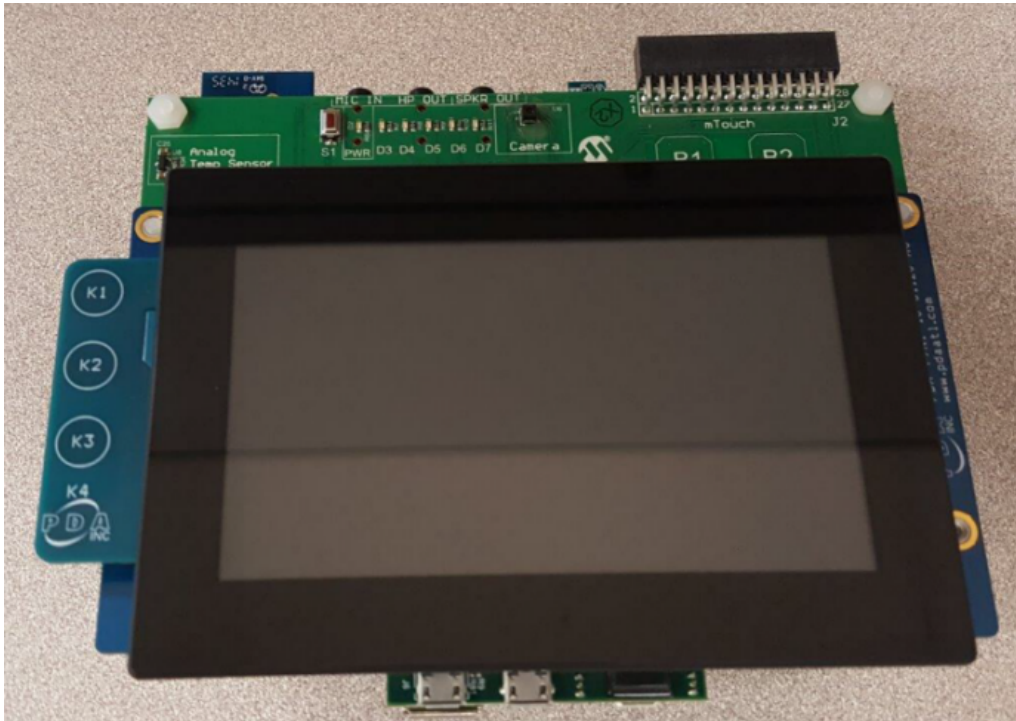
PIC32MZ Embedded Graphics with External DRAM (DA) Starter Kit plus MEB II BSP.

Description

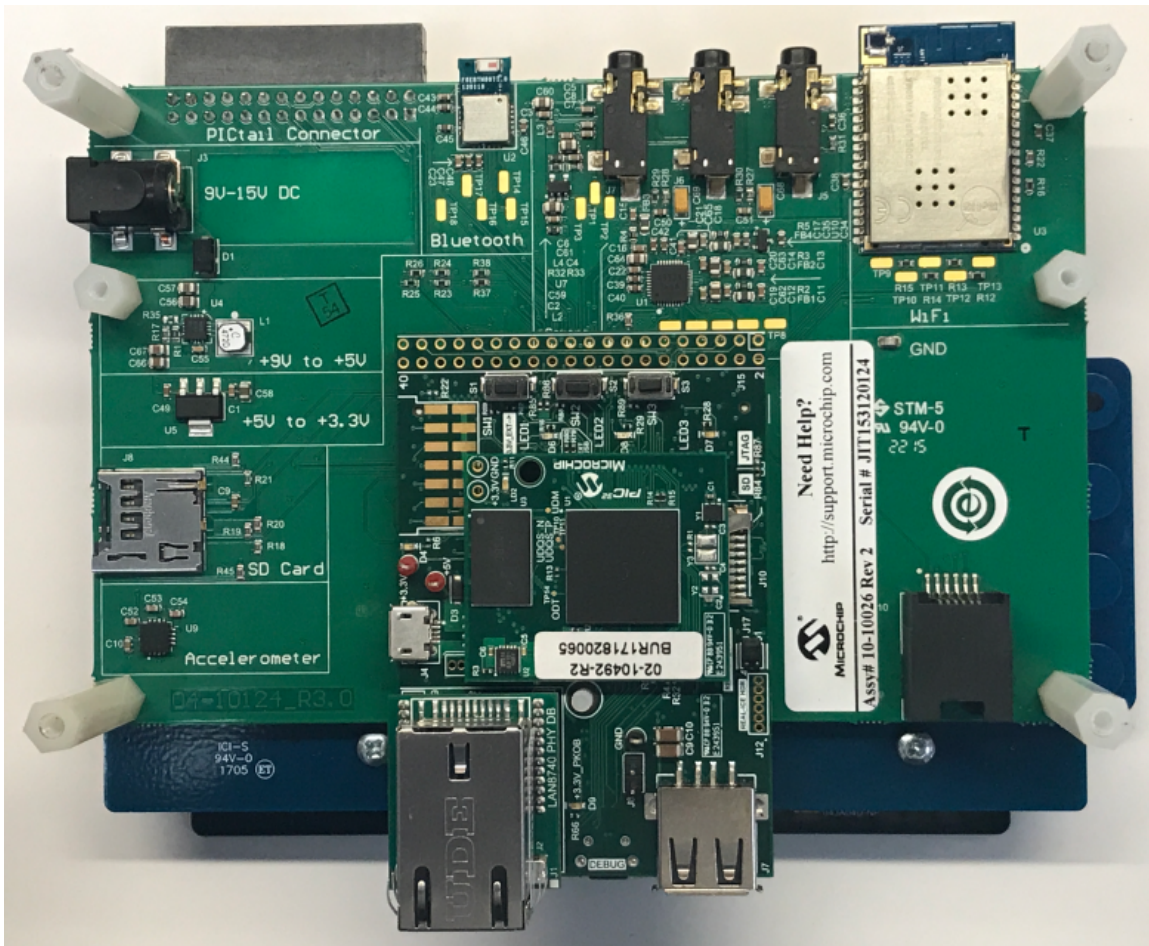
This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) connected to the [PIC32MZ Embedded Graphics with External DRAM \(DA\) Starter Kit](#).

The following figures illustrate the hardware configuration.

Top View



Bottom View



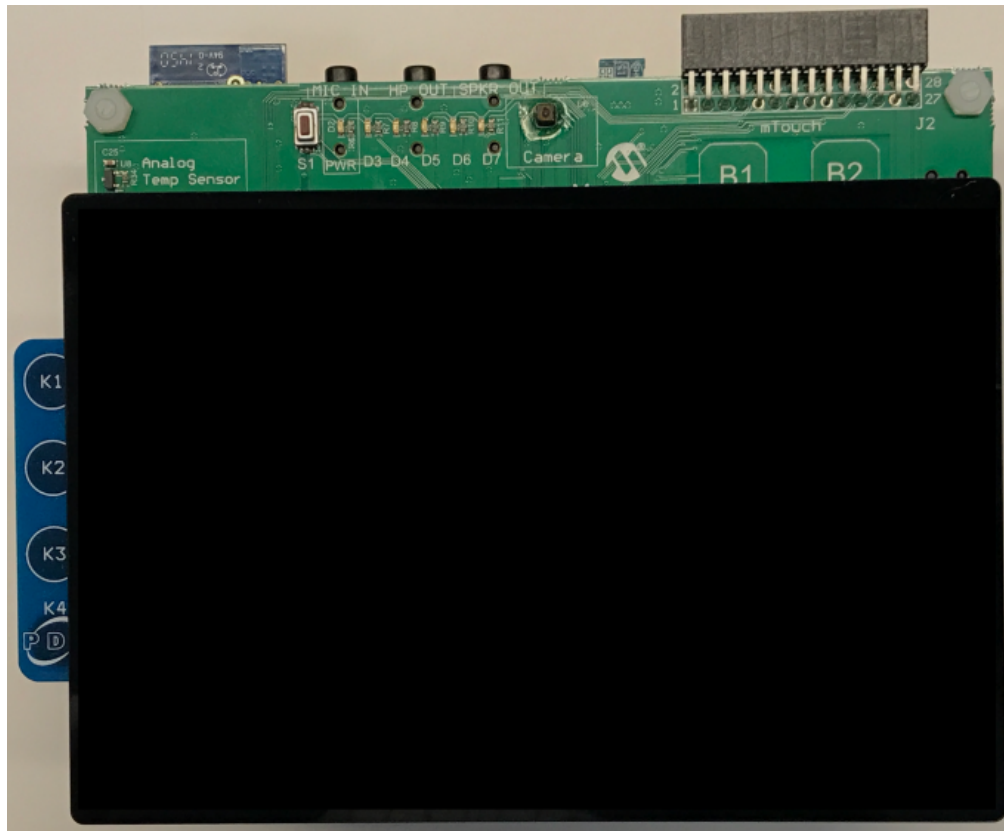
pic32mz_da_sk_extddr+meb2+vvga

PIC32MZ Embedded Graphics with External DRAM (DA) Starter Kit plus MEB II and 5" WVGA PDA Display Board BSP.

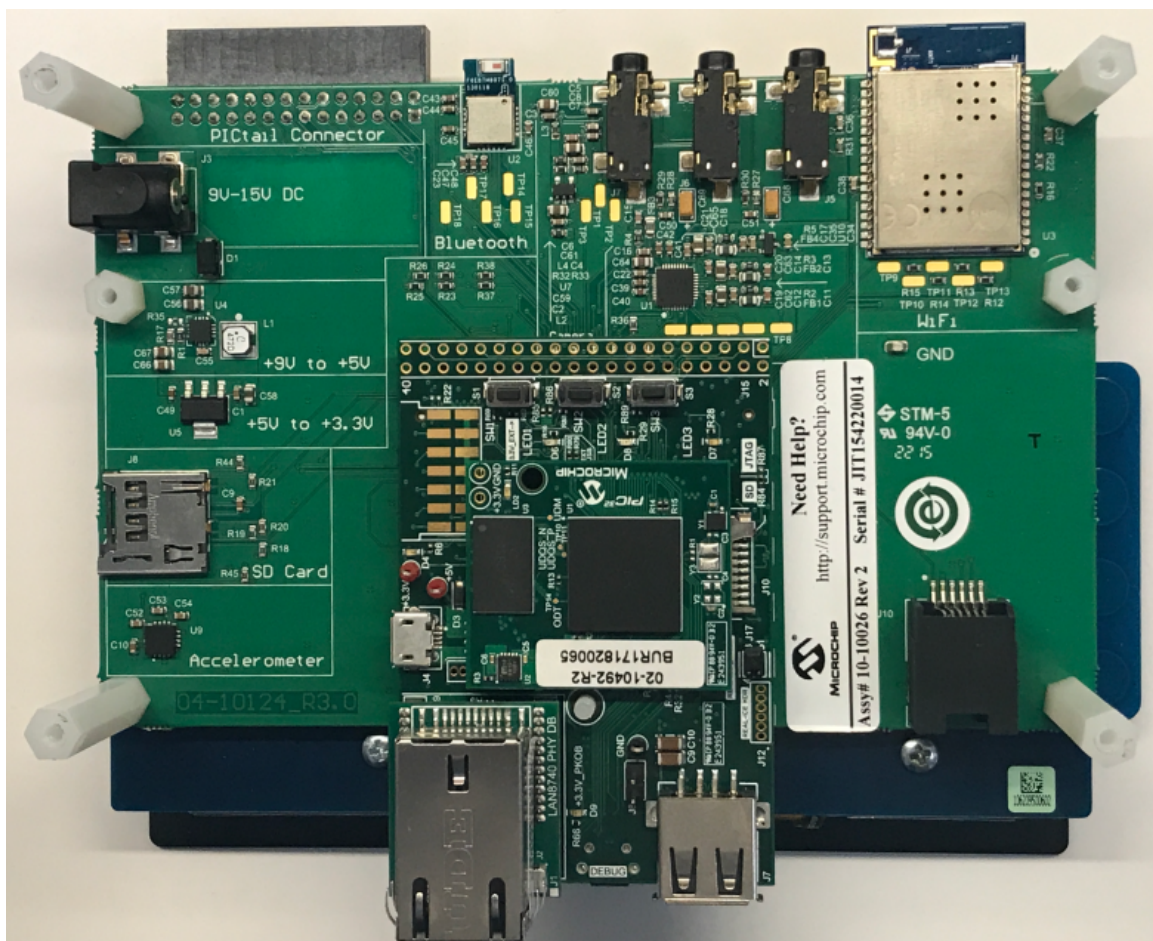
Description

This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) with the [High-Performance WVGA Display Module with maXTouch](#) connected to the [PIC32MZ Embedded Graphics with External DRAM \(DA\) Starter Kit](#).

Top View



Bottom View



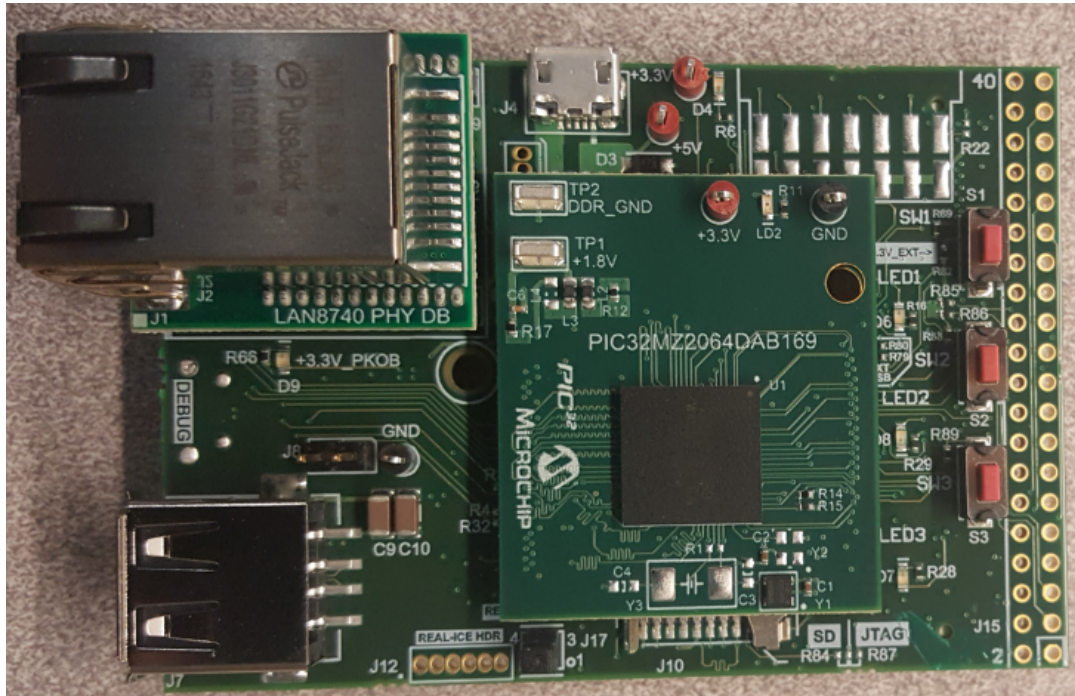
pic32mz_da_sk_intddr

PIC32MZ Embedded Graphics with Stacked DRAM (DA) Starter Kit BSP.

Description

This BSP is intended for the [PIC32MZ Embedded Graphics with External DRAM \(DA\) Starter Kit](#).

The following figure illustrates the hardware configuration.



pic32mz_da_sk_intddr+meb2

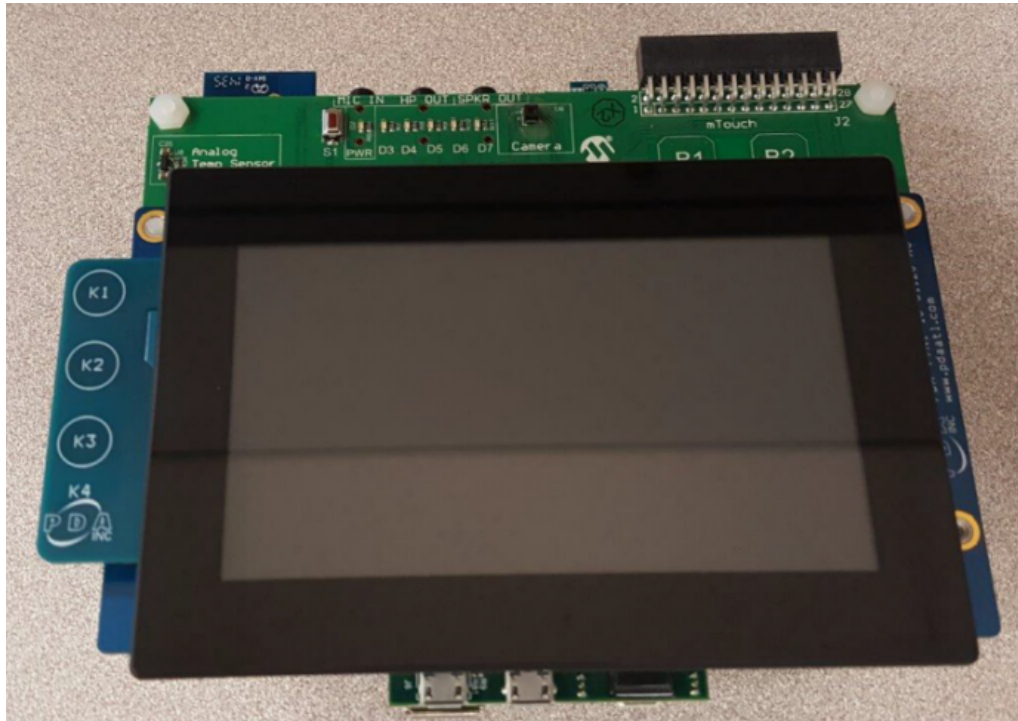
PIC32MZ Embedded Graphics with Internal DRAM (DA) Starter Kit plus MEB II BSP.

Description

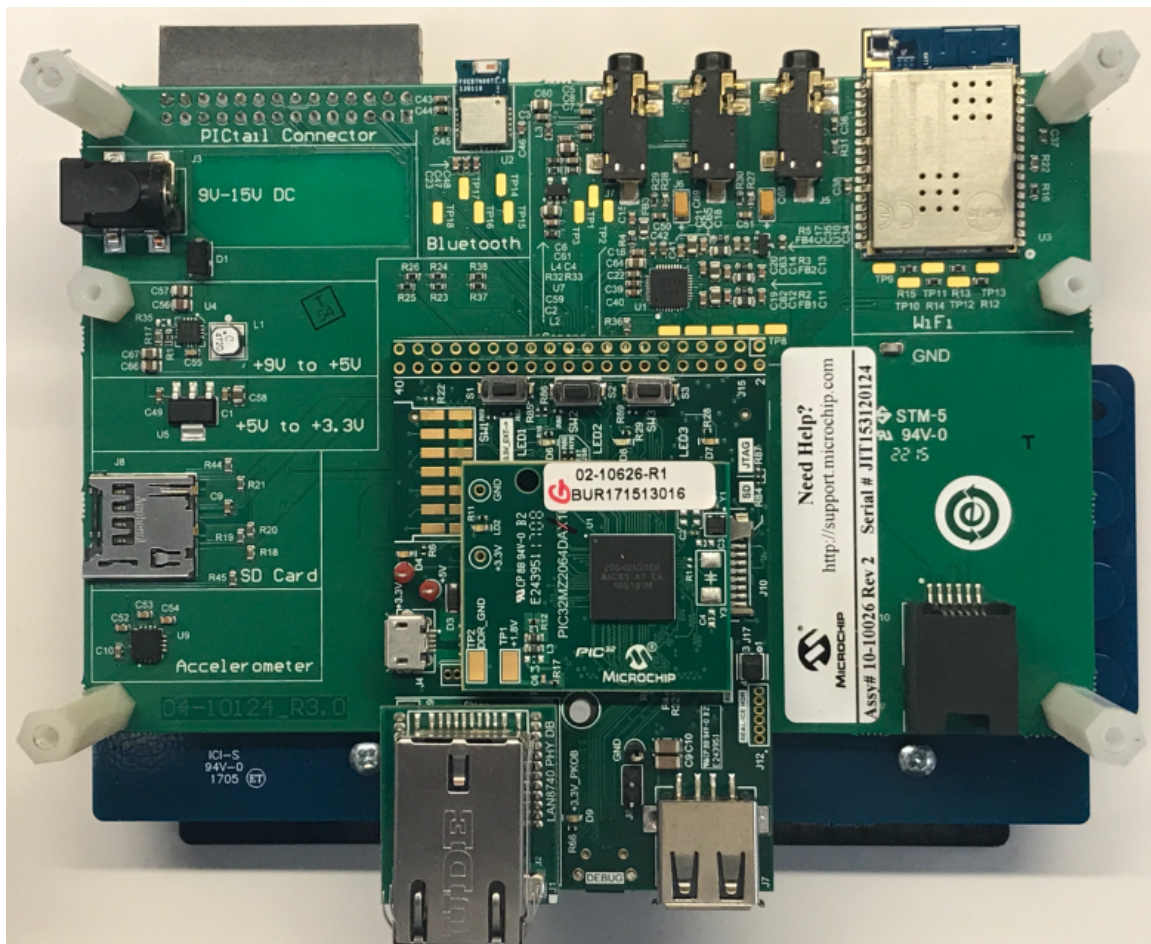
This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) connected to the [PIC32MZ Embedded Graphics with Internal DRAM \(DA\) Starter Kit](#).

The following figures illustrate the hardware configuration.

Top View



Bottom View



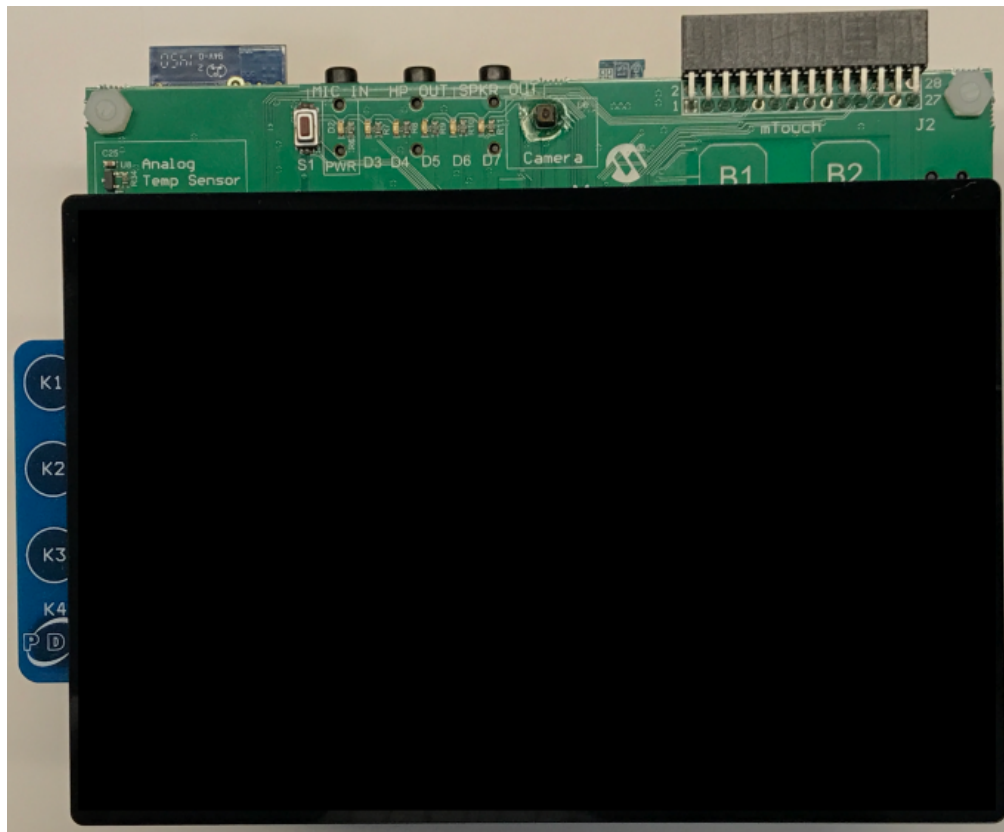
pic32mz_da_sk_intddr+meb2+wwga

PIC32MZ Embedded Graphics with Internal DRAM (DA) Starter Kit plus MEB II and 5" WVGA PDA Display Board BSP.

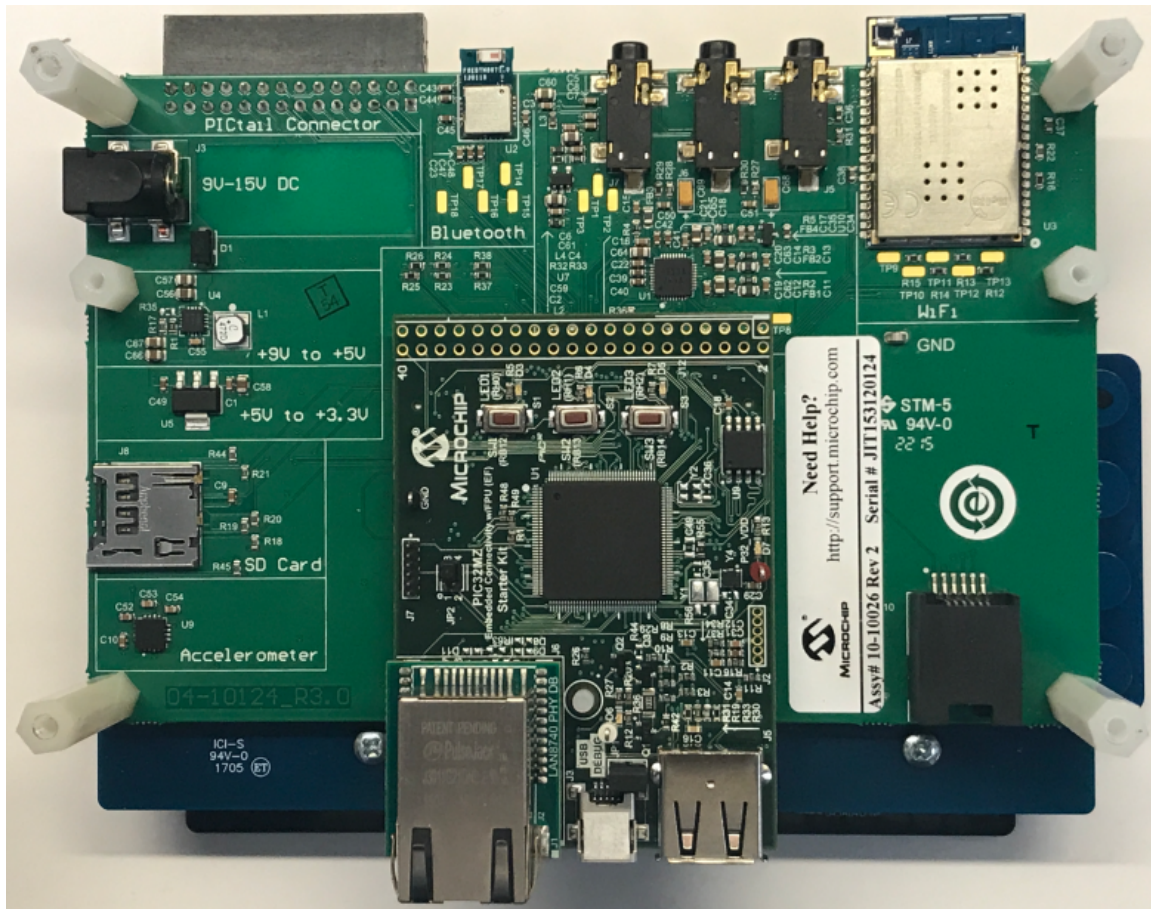
Description

This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) with the [High-Performance WVGA Display Module with maXTouch](#) connected to the [PIC32MZ Embedded Graphics with Internal DRAM \(DA\) Starter Kit](#).

Top View



Bottom View



pic32mz_da_sk_noddr+meb2

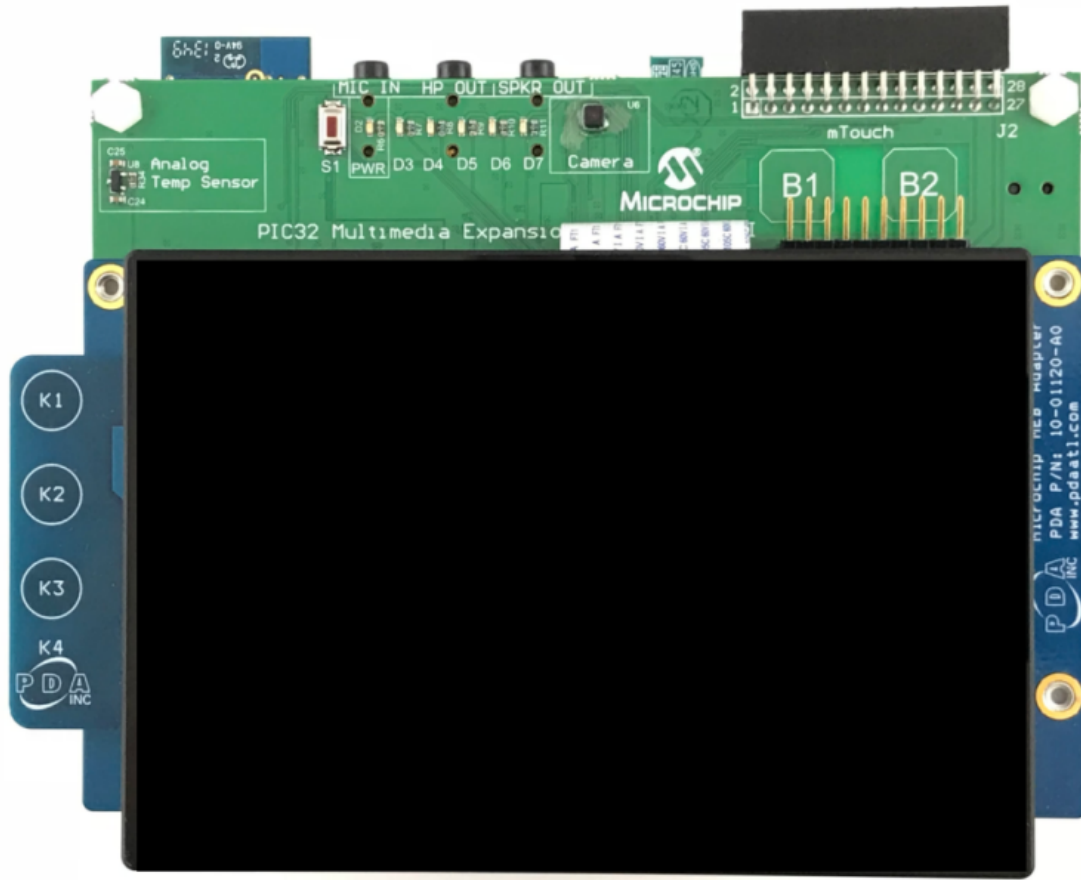
PIC32MZ Embedded Graphics with Disabled DRAM (DA) Starter Kit plus Multimedia Expansion Board II (MEB II) BSP.

Description

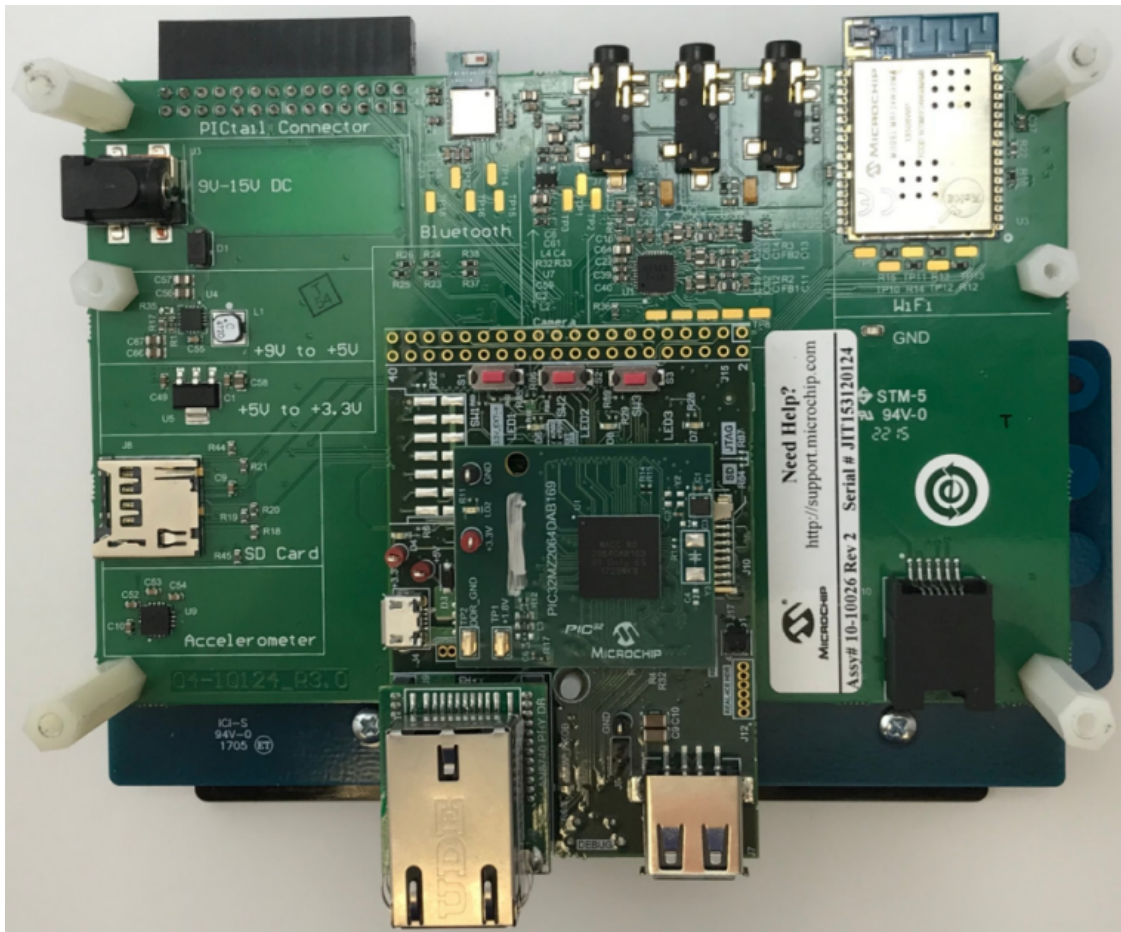
This BSP is intended for the [Multimedia Expansion Board II](#) (MEB II) connected to the [PIC32MZ Embedded Graphics with Disabled DRAM \(DA\) Starter Kit](#).

The following figures illustrate the hardware configuration.

Top View



Bottom View



pic32mz_da_sk_noddr+meb2+vvga

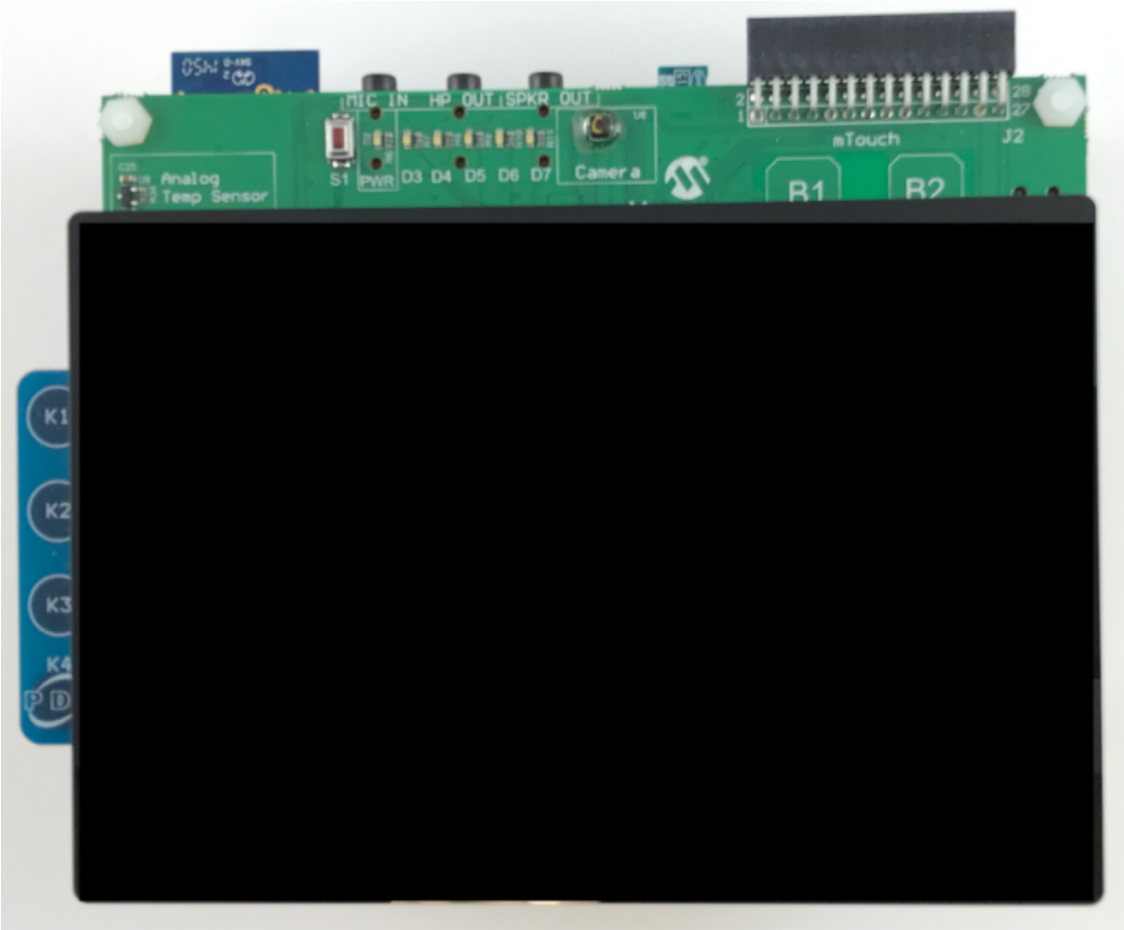
PIC32MZ Embedded Graphics with Disabled DRAM (DA) Starter Kit plus MEB II and 5" WVGA PCAP Display Board BSP.

Description

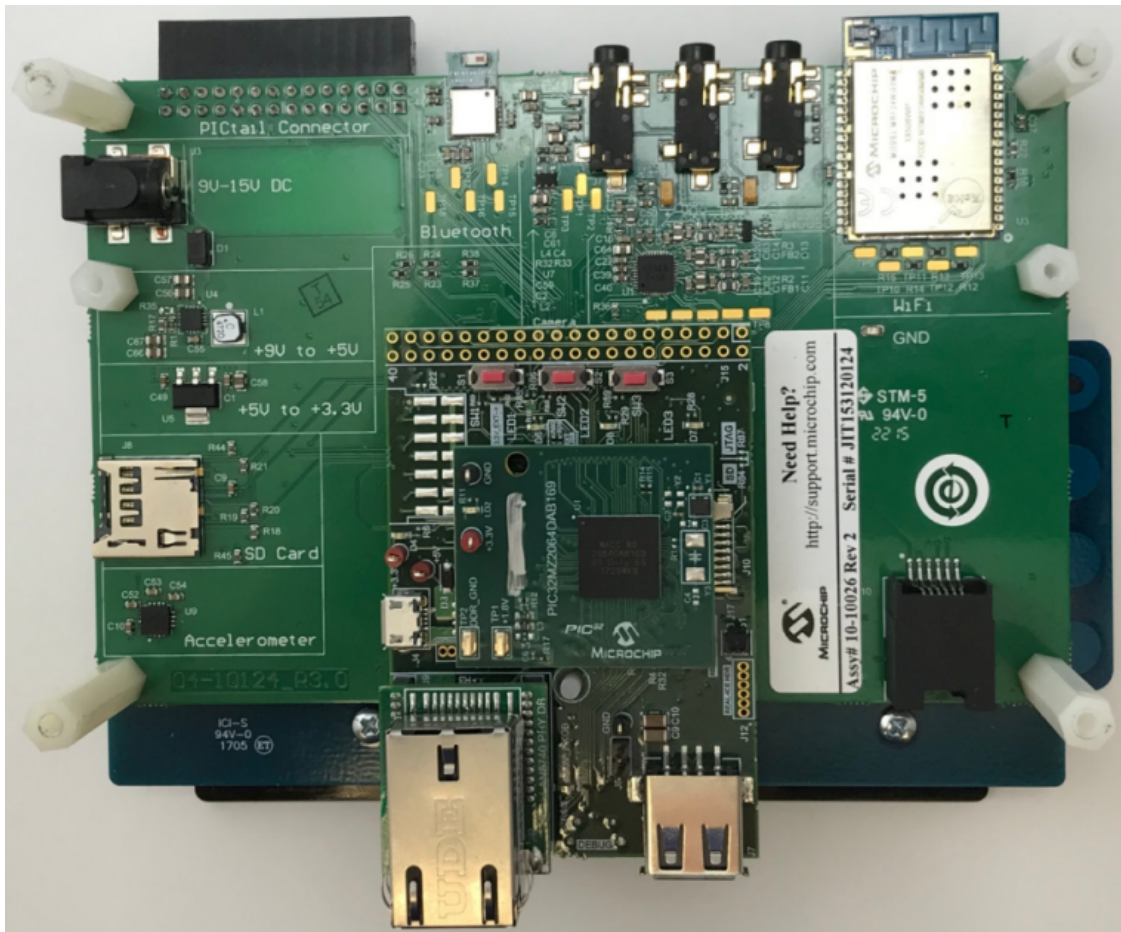
This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) with the [High-Performance WVGA Display Module with maXTouch](#) connected to the [PIC32MZ Embedded Graphics with Disabled DRAM \(DA\) Starter Kit](#).

The following figures illustrate the hardware configuration.

Top View



Bottom View



pic32mz_ec_pim+bt_audio_dk

PIC32MZ2048ECH144 Audio Plug-in Module (PIM) plus PIC32 Bluetooth Audio Development Kit BSP.

Description

This BSP is intended for the [PIC32MZ2048ECH144 Audio Plug-in Module \(PIM\)](#) connected to the [PIC32 Bluetooth Audio Development Kit](#). The following figure illustrates the hardware configuration.

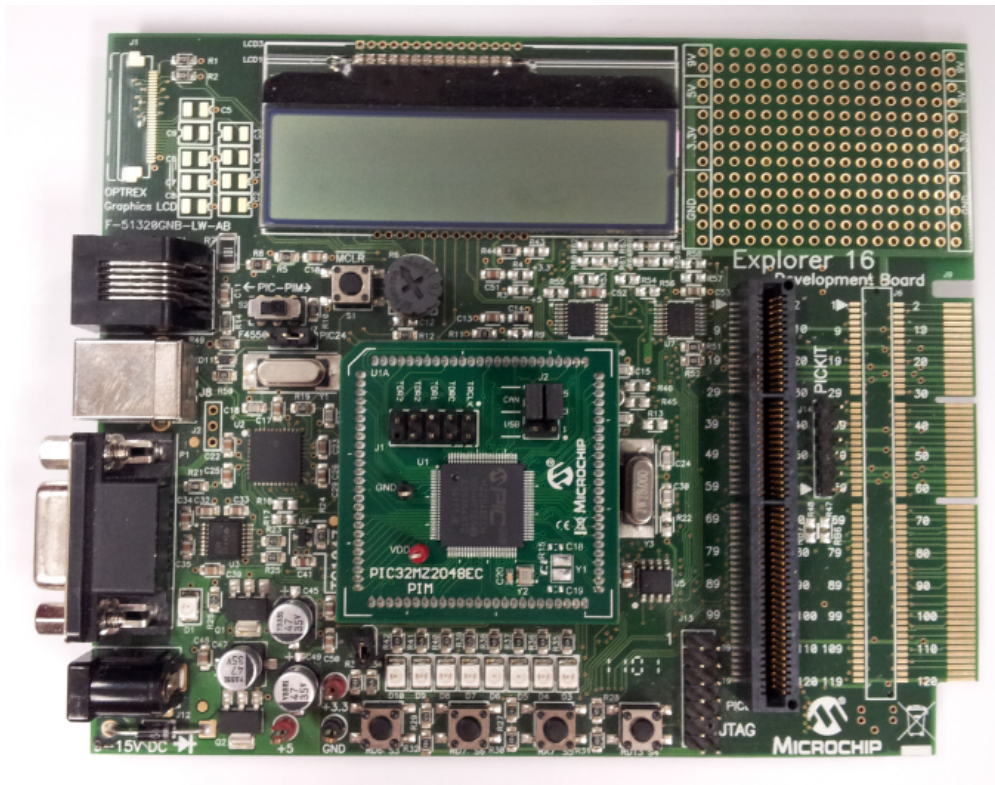


`pic32mz_ec_pim+e16`

PIC32MZ2048ECH100 Plug-in Module (PIM) plus Explorer 16 Development Board BSP.

Description

This BSP is intended for the [PIC32MZ2048ECH100 Plug-in Module \(PIM\)](#) connected to the [Explorer 16 Development Board](#). The following figure illustrates the hardware configuration.



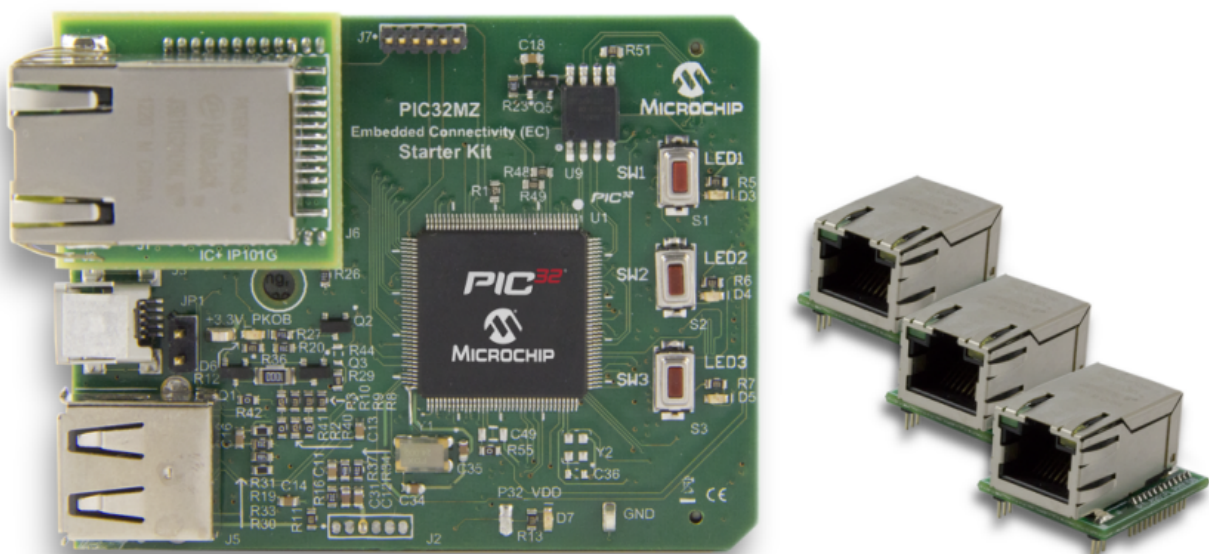
pic32mz_ec_sk

PIC32MZ EC Starter Kit BSP.

Description

This BSP is intended for the [PIC32MZ Embedded Connectivity \(EC\) Starter Kit](#).

The following figure illustrates the hardware configuration.

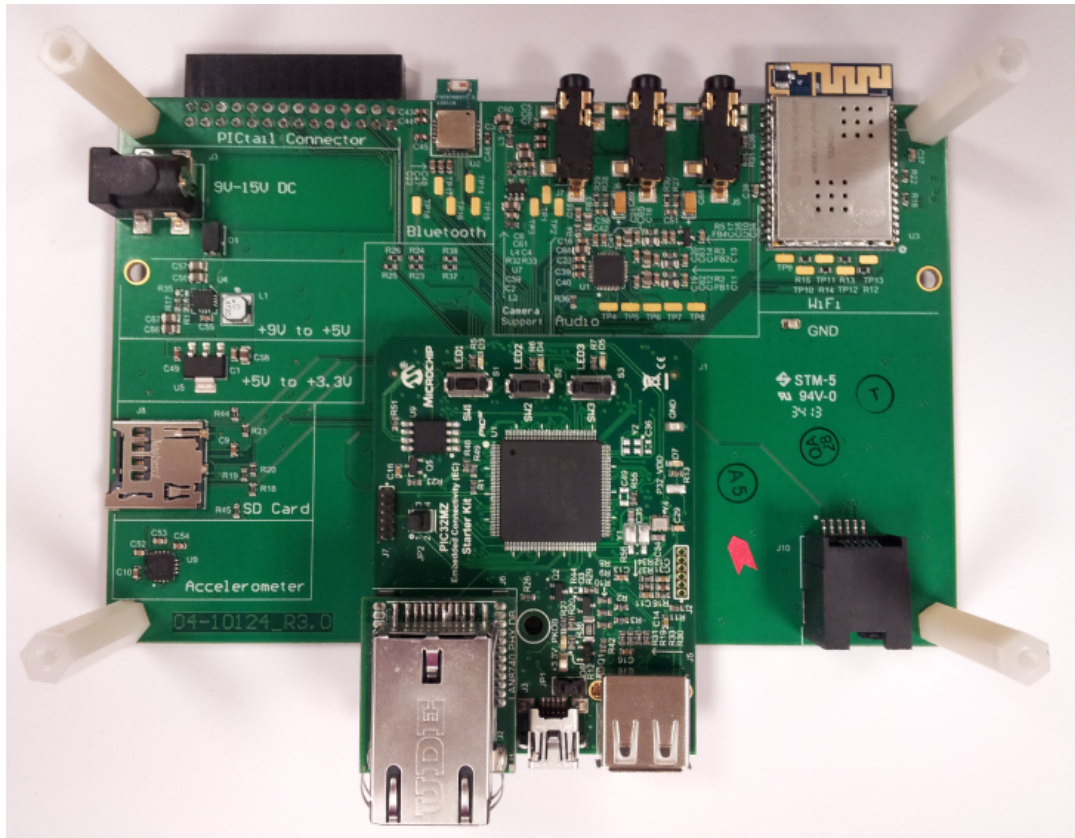


pic32mz_ec_sk+meb2

PIC32MZ EC Starter Kit plus MEB II BSP.

Description

This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) with the [PIC32MZ Embedded Connectivity \(EC\) Starter Kit](#). The following figure illustrates the hardware configuration.



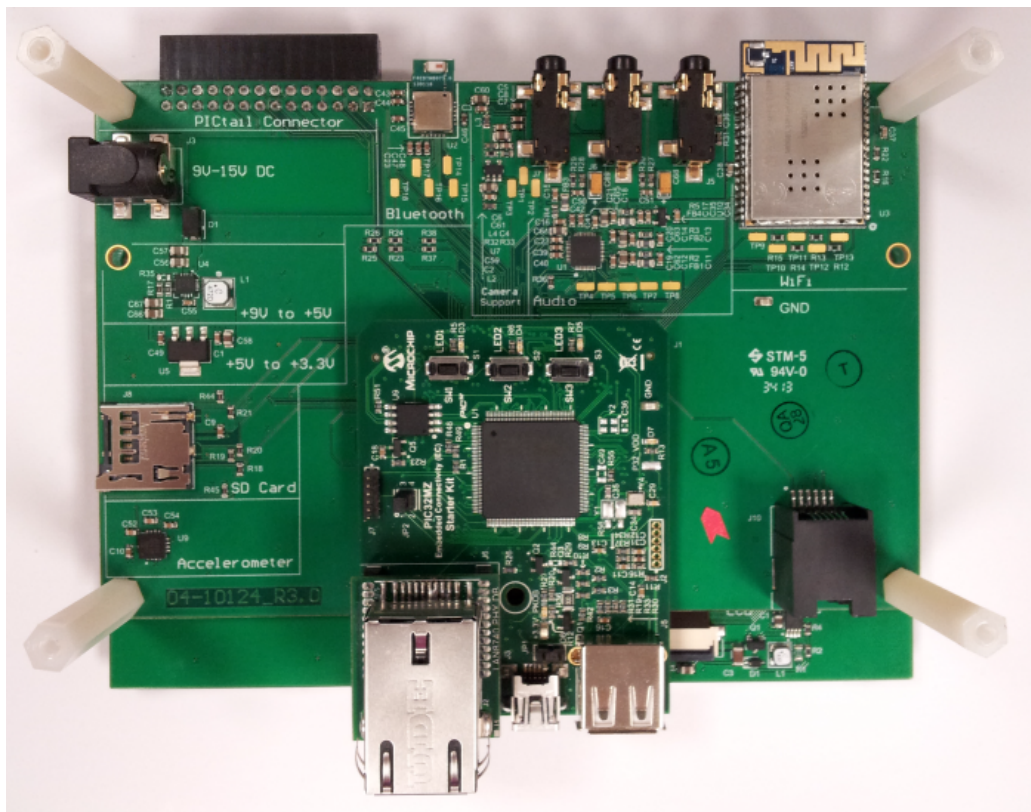
pic32mz_ec_sk+meb2+wvga

PIC32MZ EC Starter Kit plus MEB II and 5" WVGA PCAP Display Board BSP.

Description

This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) with the [High-Performance WVGA Display Module with maXTouch](#) connected to the [PIC32MZ Embedded Connectivity \(EC\) Starter Kit](#).

The following figure illustrates the hardware configuration.



pic32mz_ec_sk+s1d_pictail+vga

PIC32MZ EC Starter Kit plus Graphics Controller Pictail Plus Epson S1D13517 Daughter Board with Graphics Display Truly 5.7" 640x480 Board.

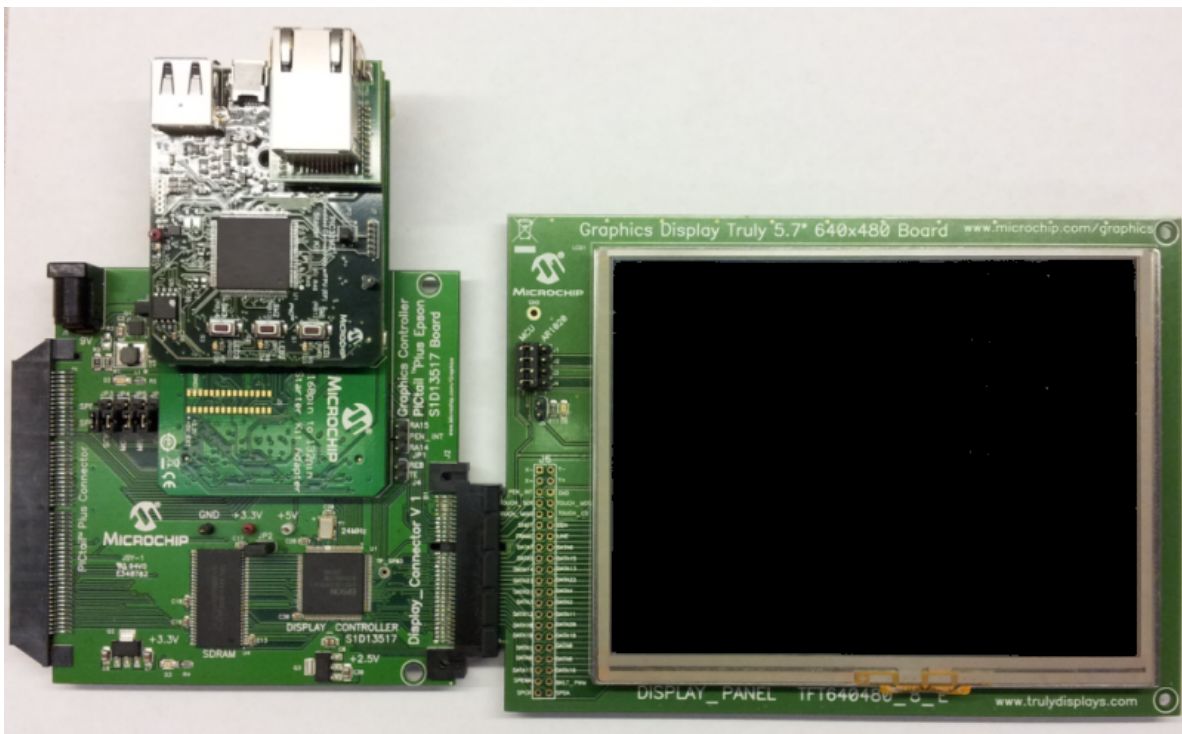
Description

This BSP is intended for the [Graphics Controller Pictail Plus Epson S1D13517 Daughter Board](#) with the [Graphics Display Truly 5.7" 640x480 Board](#) connected to the [PIC32MZ Embedded Connectivity \(EC\) Starter Kit](#).



Note:

The starter kit shown in the following figure is the PIC32MZ EF Starter Kit. The PIC32MZ EC and PIC32MZ EF starter kits are identical with the exception of the on-board device, so the hardware configuration is the same regardless of which starter kit is used.



pic32mz_ec_sk+s1d_pictail+wqvga

PIC32MZ EC Starter Kit plus Graphics Controller PICTail Plus Epson S1D13517 Daughter Board with Graphics Display Powertip 4.3" 480x272 Board BSP.

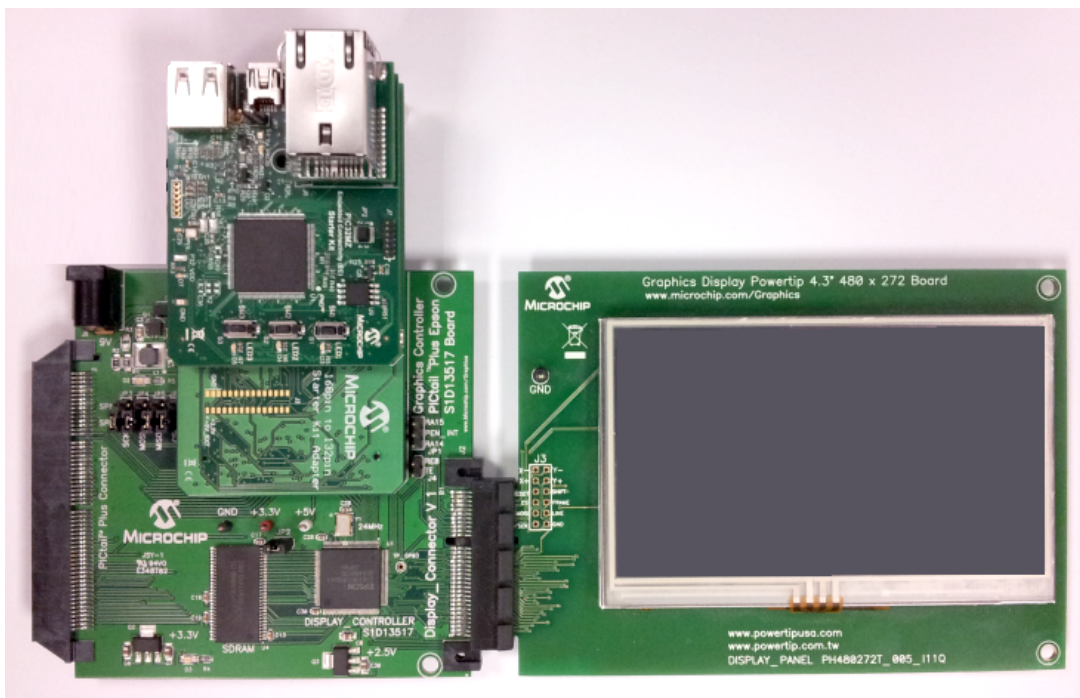
Description

This BSP is intended for the [Graphics Controller PICTail Plus Epson S1D13517 Daughter Board](#) with the [Graphics Display Powertip 4.3" 480x272 Board](#) connected to the [PIC32MZ Embedded Connectivity \(EC\) Starter Kit](#) with the [PIC32MZ Starter Kit Adapter Board](#).



Note: The PIC32MZ EC Adapter Board is required when using the Graphics Controller PICTail Plus Epson S1D13517 Daughter Board with the PIC32MZ EC Starter Kit.

The following figure illustrates the hardware configuration.

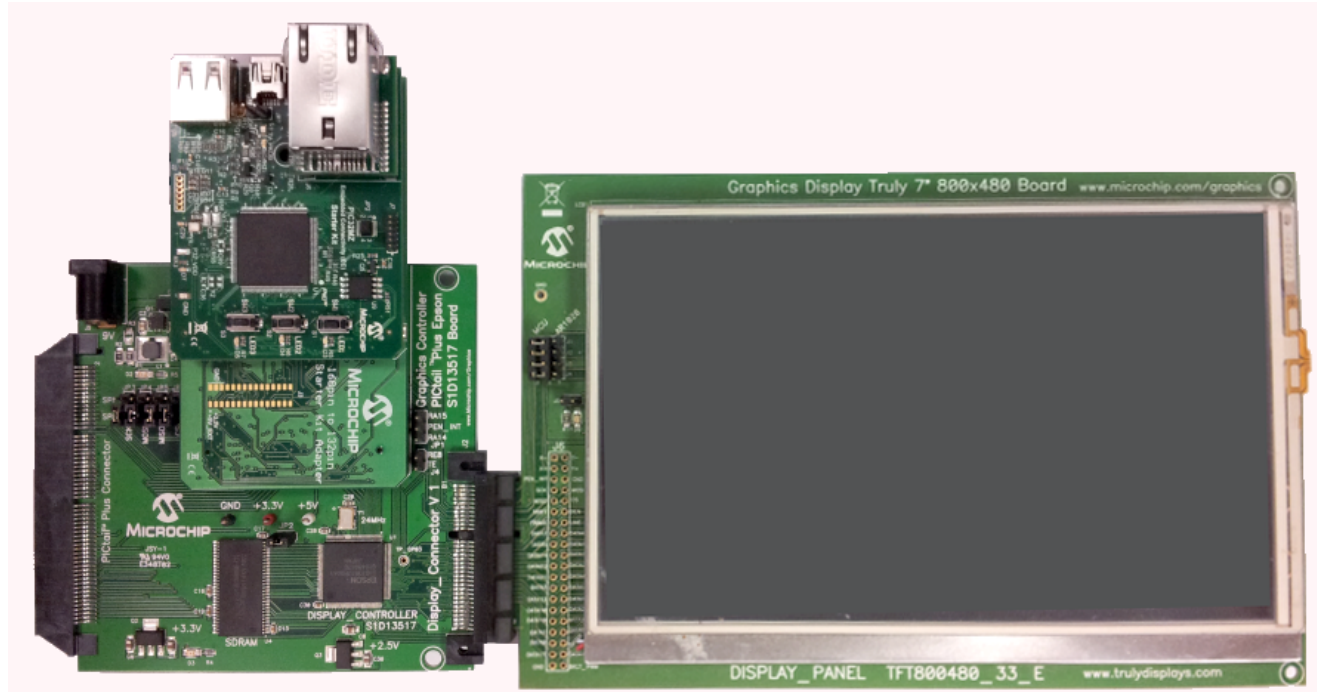


pic32mz_ec_sk+s1d_pictail+wvga

PIC32MZ EC Starter Kit plus Graphics Controller PICtail Plus Epson S1D13517 Daughter Board and Graphics Display Truly 7" 800x480 Board BSP.

Description

This BSP is intended for the [Graphics Controller PICtail Plus Epson S1D13517 Daughter Board](#) with [Graphics Display Truly 7" 800x480 Board](#) connected to the [PIC32MZ Embedded Connectivity \(EC\) Starter Kit](#).



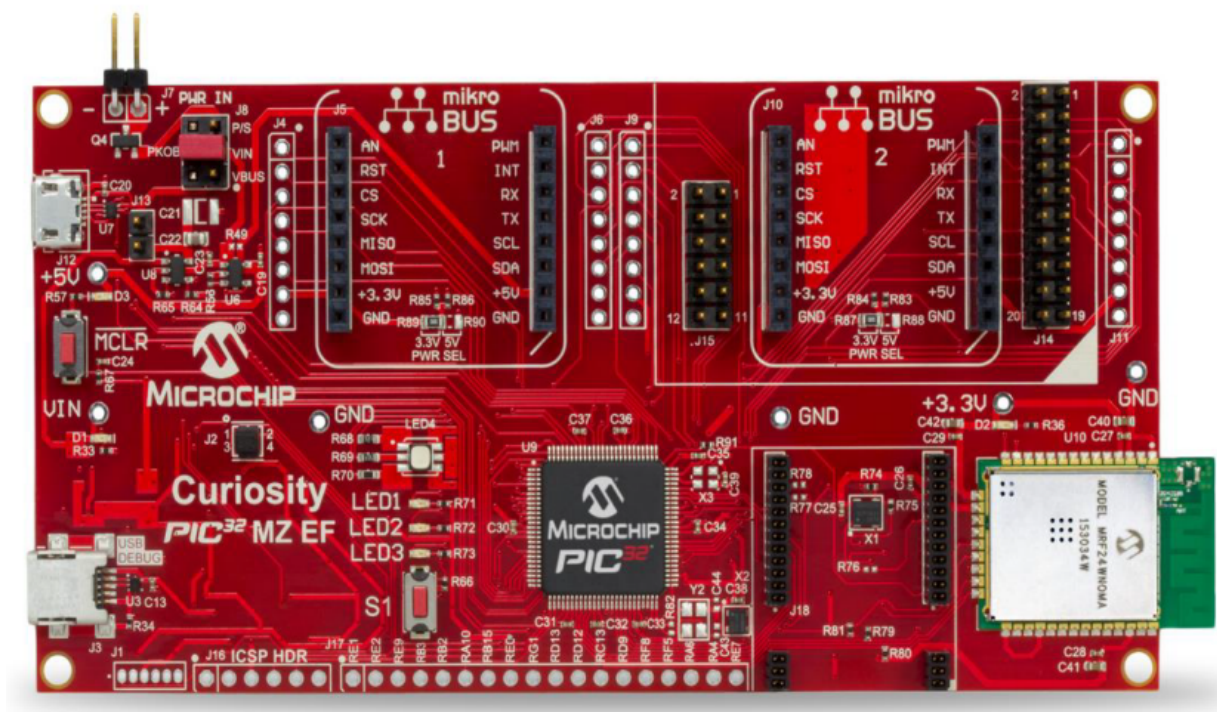
pic32mz_ef_curiosity

PIC32MZ EF Curiosity Development Board BSP.

Description

This BSP is intended for the [PIC32MZ EF Curiosity Development Board](#).

The following figure illustrates the hardware configuration.



pic32mz_ef_pim+bt_audio_dk

PIC32MZ2048EFH144 Audio Plug-in Module (PIM) plus PIC32 Bluetooth Audio Development Kit BSP.

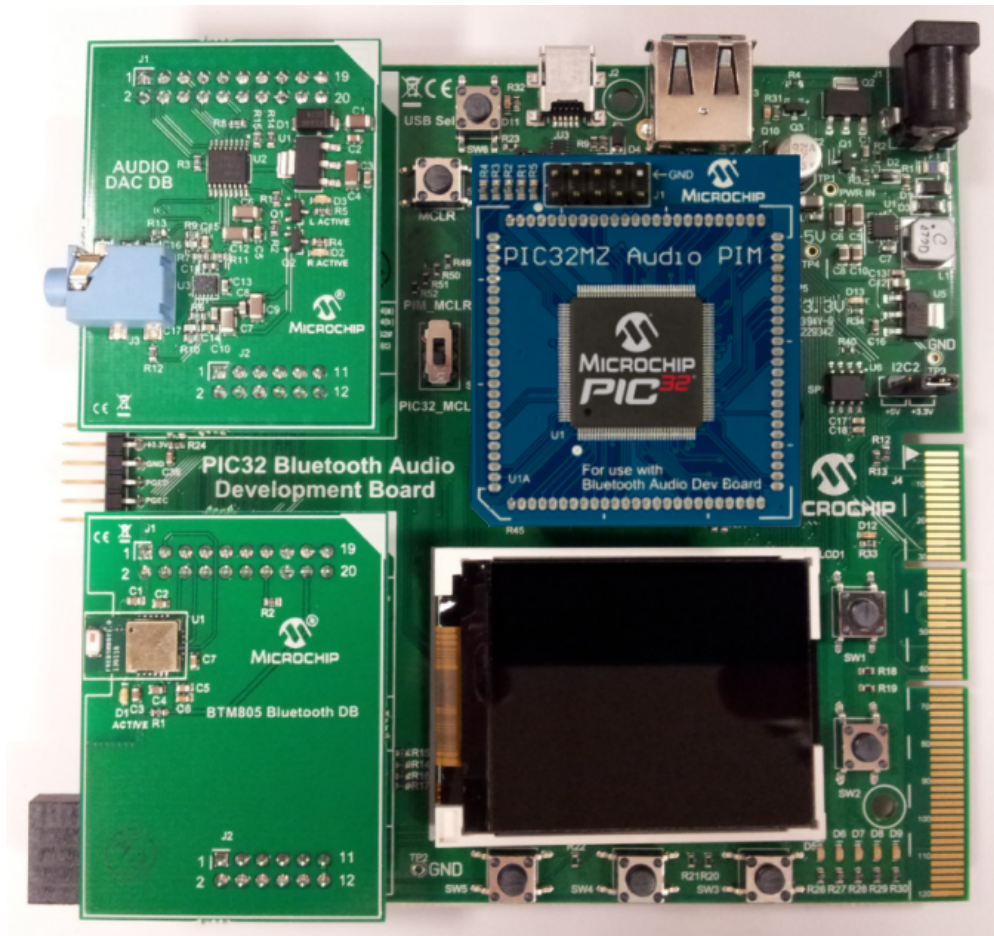
Description

This BSP is intended for the [PIC32MZ2048EFH144 Audio Plug-in Module \(PIM\)](#) connected to the [PIC32 Bluetooth Audio Development Kit](#).



Note:

The PIM shown in the following figure is the PIC32MZEC2048. The PIC32MZ EC and PIC32MZ EF PIMs are identical with the exception of the on-board device, so the hardware configuration is the same regardless of which PIM is used.

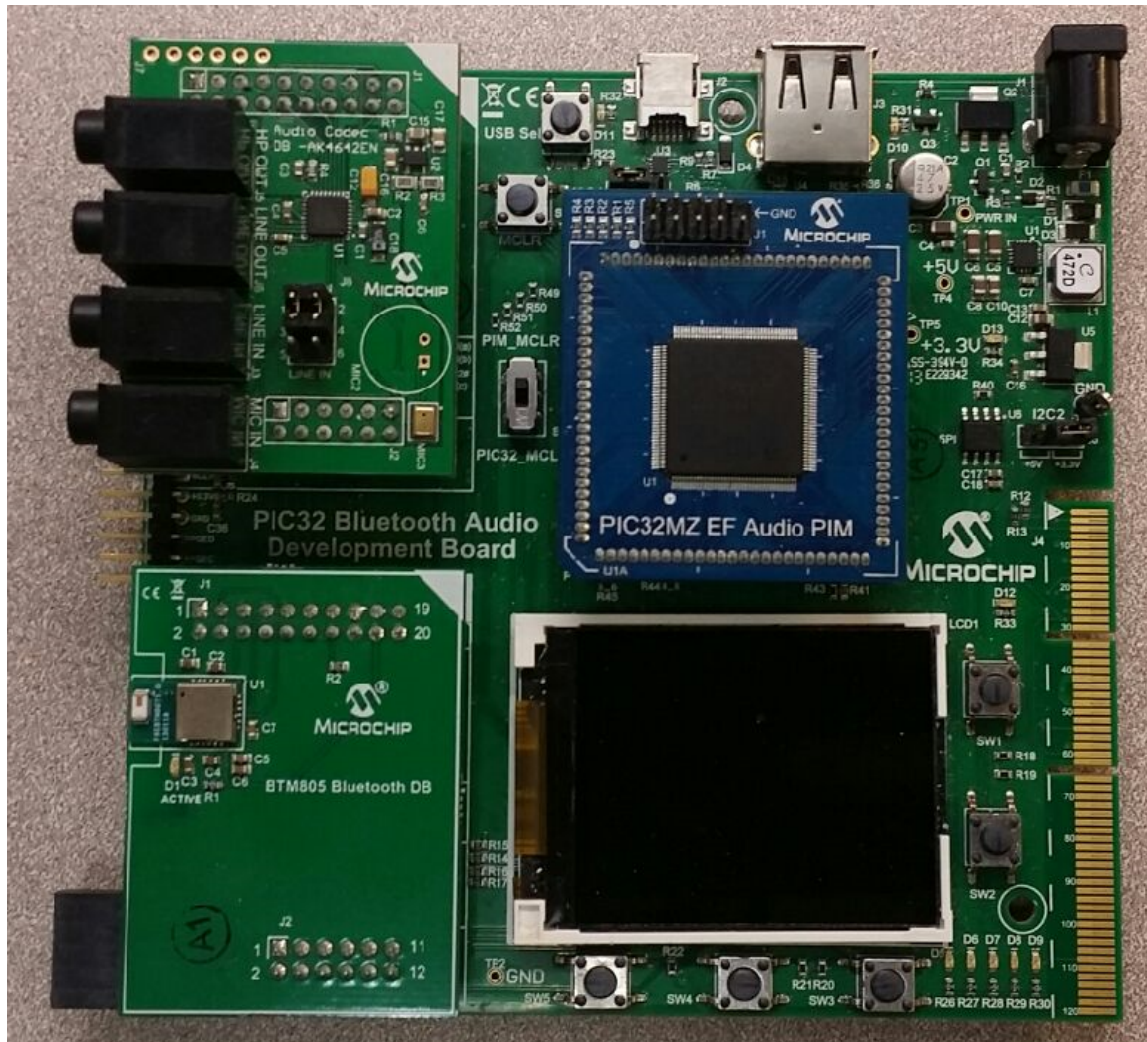


pic32mz_ef_pim+bt_audio_dk+ak4642

PIC32MZ2048EFH144 Audio Plug-in Module (PIM) plus PIC32 Bluetooth Audio Development Kit and AK4642 Audio Codec BSP.

Description

This BSP is intended for the [PIC32MZ2048EFH144 Audio Plug-in Module \(PIM\)](#) connected to the [PIC32 Bluetooth Audio Development Kit](#) with the AK4642 Audio Codec.



pic32mz_ef_pim+e16

PIC32MZ2048EFH100 Plug-in Module (PIM) plus Explorer 16 Development Board BSP.

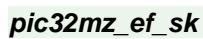
Description

This BSP is intended for the [PIC32MZ2048EFH100 Plug-in Module \(PIM\)](#) connected to the [Explorer 16 Development Board](#).



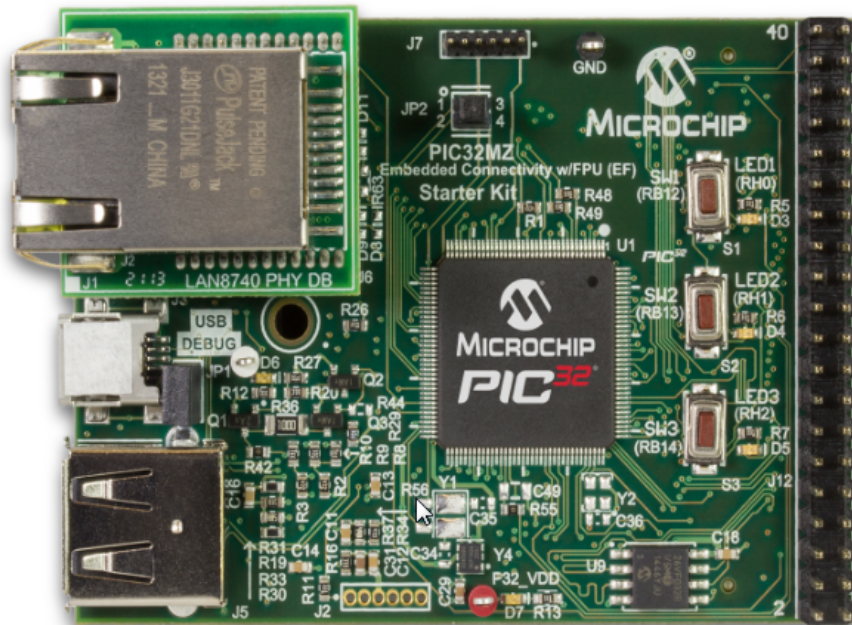
Note:

The PIM shown in the following figure is the PIC32MZEC2048. The PIC32MZ EC and PIC32MZ EF PIMs are identical with the exception of the on-board device, so the hardware configuration is the same regardless of which PIM is used.



Description

The following figure illustrates the hardware configuration.



pic32mz_ef_sk+maxtouch_xplained_pro_3_5

PIC32MZ Embedded Connectivity with Floating Point Unit (EF) Starter Kit with the maXTouch Xplained Pro 3.5" display BSP.

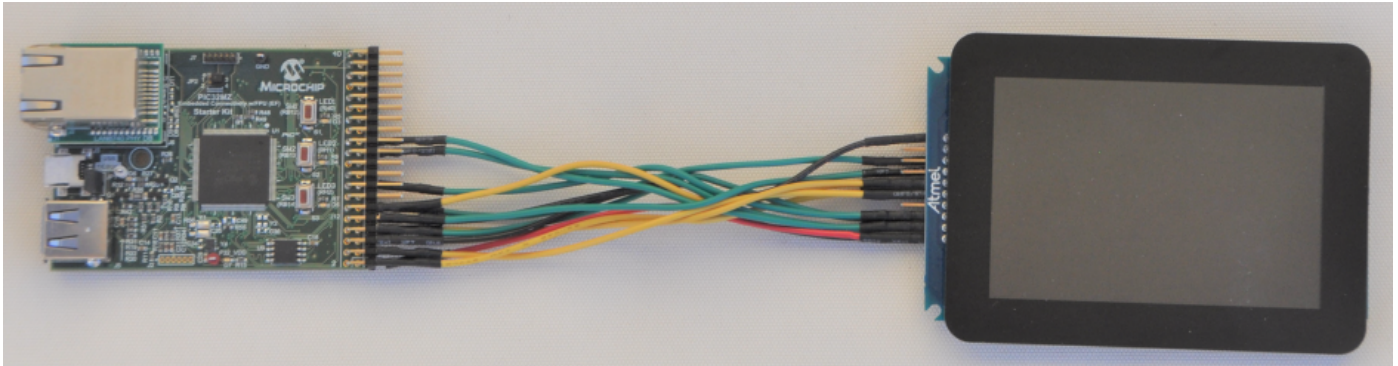
Description

This BSP is intended for the [PIC32MZ Embedded Connectivity \(EF\) Starter Kit](#) the [maXTouch Xplained Pro 3.5"](#) display. The following figure illustrates the hardware configuration.



Note:

Refer to the [aria_quickstart](#) demonstration for detailed information on pin connections between the starter kit and the display.



pic32mz_ef_sk+meb2

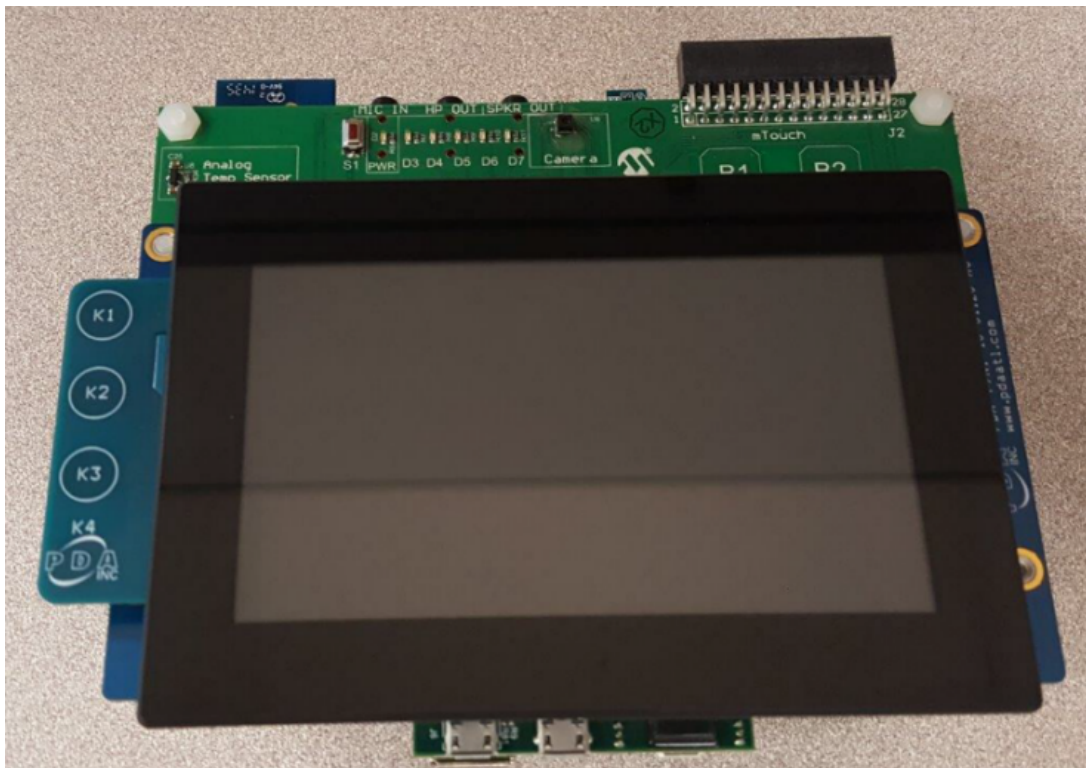
PIC32MZ EF Starter Kit plus MEB II BSP.

Description

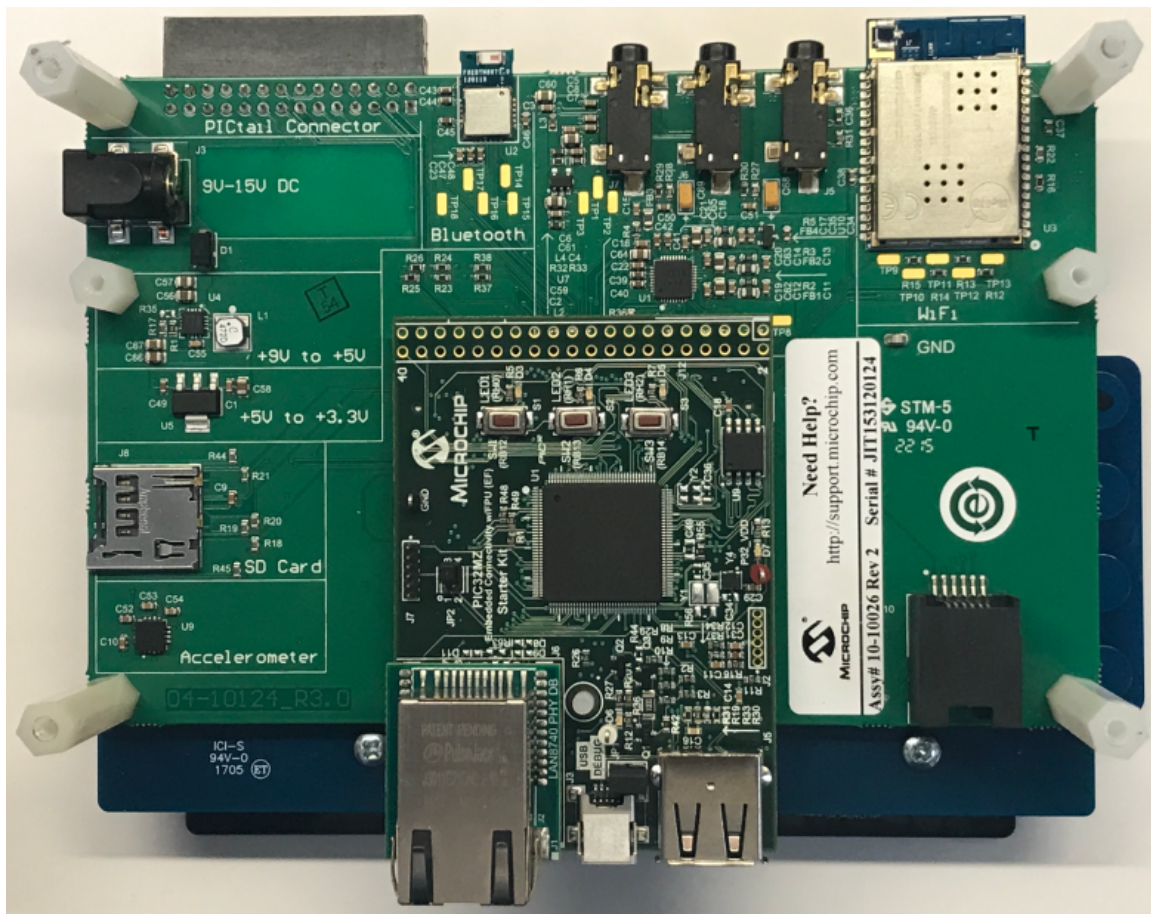
This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) connected to the [PIC32MZ Embedded Connectivity with Floating Point Unit \(EF\) Starter Kit](#).

The following figure illustrates the hardware configuration.

Top View



Bottom View



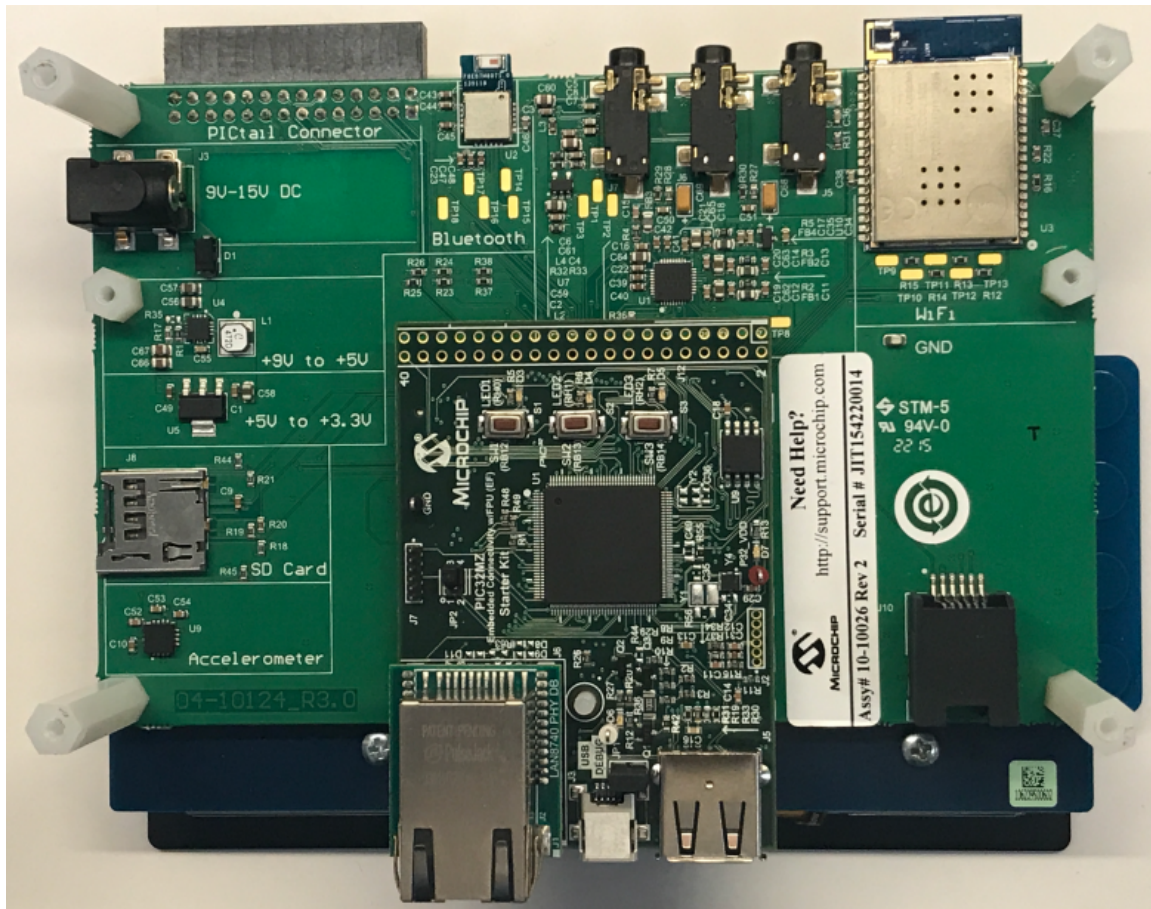
pic32mz_ef_sk+meb2+wgva

PIC32MZ EF Starter Kit plus MEB II and 5" WVGA PDA Display Board BSP.

Description

This BSP is intended for the [Multimedia Expansion Board II \(MEB II\)](#) with the [High-Performance WVGA Display Module with maXTouch](#) connected to the [PIC32MZ Embedded Connectivity with Floating Point Unit \(EF\) Starter Kit](#).

Bottom View



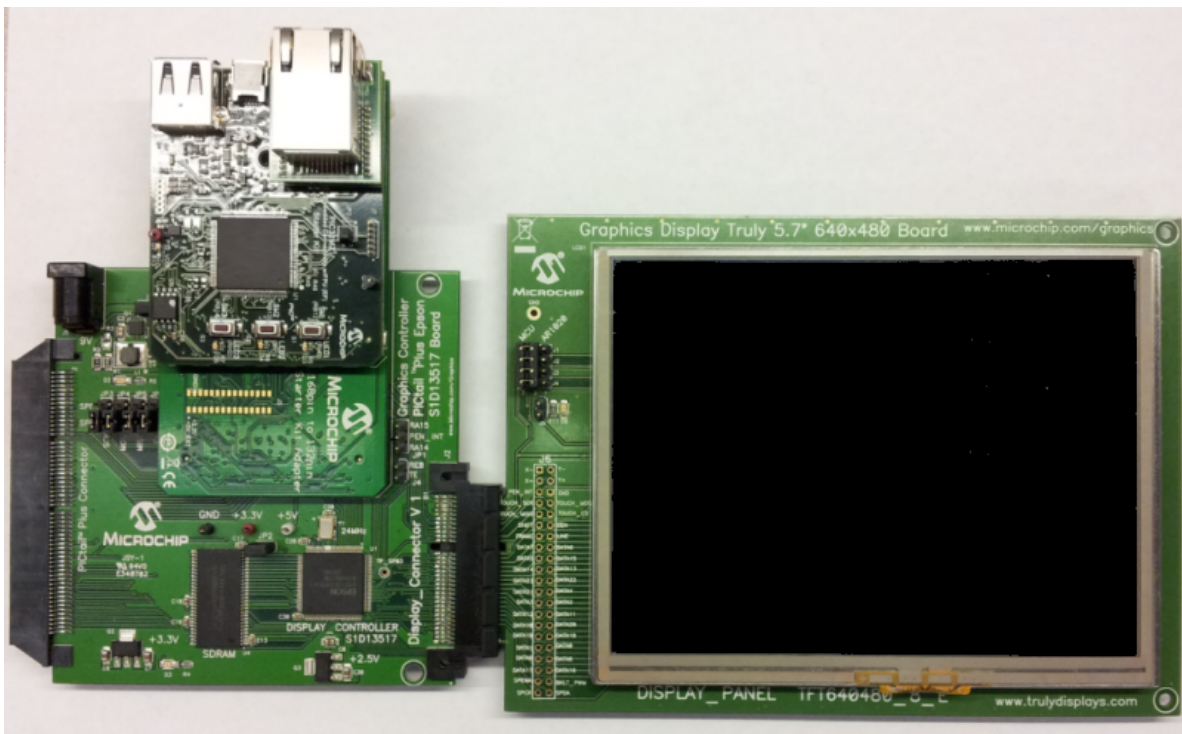
pic32mz_ef_sk+s1d_pictail+vga

PIC32MZ EF Starter Kit plus Graphics Controller PICtail Plus Epson S1D13517 Daughter Board with Graphics Display Truly 5.7" 640x480 Board.

Description

This BSP is intended for the [Graphics Controller PICtail Plus Epson S1D13517 Daughter Board](#) with the [Graphics Display Truly 5.7" 640x480 Board](#) connected to the [PIC32MZ Embedded Connectivity with Floating Point Unit \(EF\) Starter Kit](#).

The following figure illustrates the hardware configuration.



`pic32mz_ef_sk+s1d_pictail+wqvga`

PIC32MZ EF Starter Kit plus Graphics Controller PICtail Plus Epson S1D13517 Daughter Board with Graphics Display Powertip 4.3" 480x272 Board BSP.

Description

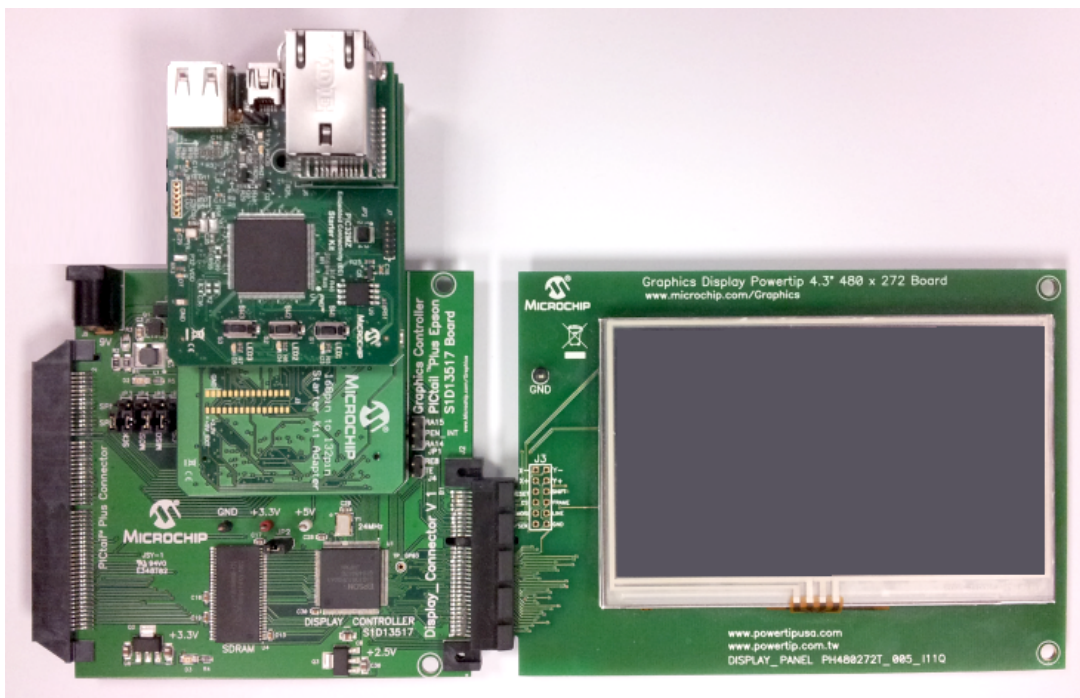
This BSP is intended for the [Graphics Controller PICtail Plus Epson S1D13517 Daughter Board](#) with the [Graphics Display Powertip 4.3" 480x272 Board](#) connected to the [PIC32MZ Embedded Connectivity with Floating Point Unit \(EF\) Starter Kit](#) with the [PIC32MZ Starter Kit Adapter Board](#).



Note:

The starter kit shown in the following figure is the PIC32MZ EC Starter Kit. The PIC32MZ EC and PIC32MZ EF starter kits are identical with the exception of the on-board device, so the hardware configuration is the same regardless of which starter kit is used.

The following figure illustrates the hardware configuration.



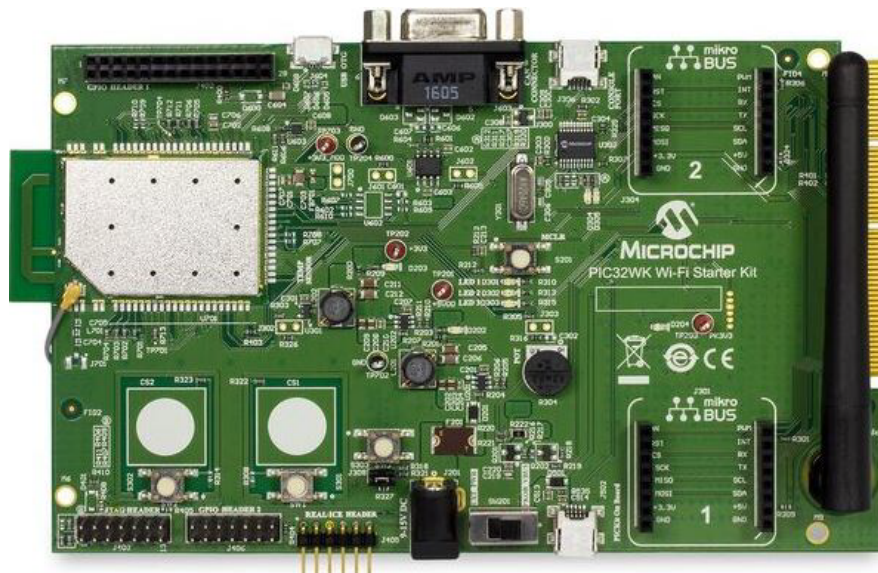
pic32wk_gbp_gpd_sk+module

PIC32WK Wi-Fi Starter Kit BSP.

Description

This BSP is intended for the PIC32WK Wi-Fi Starter Kit.

The following figure illustrates the hardware configuration.



Library Interface

Provides information on BSP functions, structs, types, and macros.

Custom Named GPIO Macros

Name	Description
custom_gpio_nameOff	Turns off the custom-named GPIO pin.
custom_gpio_nameOn	Turns on the custom-named GPIO pin.
custom_gpio_nameStateGet	Gets the current value of the custom-named GPIO pin.
custom_gpio_nameStateSet	Sets a new value on the custom-named GPIO pin.
custom_gpio_nameToggle	Toggles the custom-named GPIO pin.
custom_gpio_pin_PIN	Identifies the bit position within the port channel associated with a custom-named GPIO pin.
custom_gpio_pin_PIN_MASK	Identifies the bit mask within the port channel associated with a custom-named GPIO pin.
custom_gpio_pin_PORT	Identifies the port channel associated with a custom-named GPIO pin.

Custom Named LED Macros

Name	Description
custom_led_nameOff	Turns off the custom-named LED.
custom_led_nameOn	Turns on the custom-named LED.
custom_led_nameStateGet	Gets the current state of the custom-named LED.
custom_led_nameToggle	Toggles the custom-named LED.

Custom Named Switch State Macros


Name	Description
custom_switch_nameStateGet	Gets the current value of the custom-named switch.

Data Types and Constants






Name	Description
BSP_LED	Defines the names of the LEDs available on the selected board.

	BSP_SWITCH_STATE	Defines possible states of the switches on this board.
	BSP_LED_STATE	Enumerates the supported LED states.
	BSP_SWITCH	Defines the switches available on this board.
	BSP_OSC_FREQUENCY	Defines the frequency value of crystal/oscillator used on the board


Initialization Functions

	Name	Description
	BSP_Initialize	Performs the necessary actions to initialize a board.

LED Control Functions

	Name	Description
	BSP_LEDOff	Turns OFF the specified LED.
	BSP_LEDOn	Turns on the specified LED.
	BSP_LEDStateGet	Returns the current state of the LED.
	BSP_LEDStateSet	Controls the state of the LED.
	BSP_LEDToggle	Toggles the current state of the LED.

Switch State Functions

	Name	Description
	BSP_SwitchStateGet	Returns the current state of the specified switch.

Description

The header file for each BSP library uses the file named `bsp_config.h`.

Initialization Functions

BSP_Initialize Function

Performs the necessary actions to initialize a board.

File

`bsp_help.h`

C

```
void BSP_Initialize();
```

Returns

None.

Description

This function initializes the LED and Switch ports on the board. This function must be called by the user before using any APIs present on this BSP.

Remarks

None.

Preconditions

None.

Example

```
//Initialize the BSP
BSP_Initialize();
```

Function

```
void BSP_Initialize ( void )
```

LED Control Functions

BSP_LEDOff Function

Turns OFF the specified LED.

File

bsp_help.h

C

```
void BSP_LEDOff(BSP_LED led);
```

Returns

None.

Description

This function turns OFF the specified LED.

Remarks

None.

Preconditions

[BSP_Initialize\(\)](#) should have been called.

Example

```
// Turn off LED1 on the board
BSP_LEDOff(BSP_LED_1);
```

Parameters

Parameters	Description
led	The name of the LED to turn off.

Function

```
void BSP_LEDOff ( BSP_LED led );
```

BSP_LEDOn Function

Turns on the specified LED.

File

bsp_help.h

C

```
void BSP_LEDOn(BSP_LED led);
```

Returns

None.

Description

This function turns on the specified LED.

Remarks

None.

Preconditions

[BSP_Initialize\(\)](#) should have been called.

Example

```
// Turn on LED1 on the board
BSP_LEDOn(BSP_LED_1);
```

Parameters

Parameters	Description
led	The name of the LED to turn on

Function

```
void BSP_LEDOn ( BSP_LED led );
```

BSP_LEDStateGet Function

Returns the current state of the LED.

File

bsp_help.h

C

```
BSP_LED_STATE BSP_LEDStateGet (BSP_LED led);
```

Returns

BSP_LED_STATE_OFF If the LED is off. BSP_LED_STATE_ON If the LED is on.

Description

This function returns the current state of the LED.

Remarks

None.

Preconditions

[BSP_Initialize\(\)](#) should have been called.

Example

```
// Check if LED2 is off.
if (BSP_LED_STATE_OFF == BSP_LEDStateGet (BSP_LED_2)
{
    // Turn the LED on.
    BSP_LEDStateSet (BSP_LED_2, BSP_LED_STATE_ON);
}
```

Parameters

Parameters	Description
led	The name of the LED to whose state to obtain.

Function

```
BSP_LED_STATE BSP_LEDStateGet ( BSP_LED led );
```

BSP_LEDStateSet Function

Controls the state of the LED.

File

bsp_help.h

C

```
void BSP_LEDStateSet (BSP_LED led, BSP_LED_STATE state);
```

Returns

None.

Description

This function allows the caller to specify the state of the LED.

Remarks

None.

Preconditions

`BSP_Initialize()` should have been called.

Example

```
// Turn on LED1 on the board
BSP_LEDStateSet(BSP_LED_1, BSP_LED_STATE_ON);

// Turn off LED2 on the board
BSP_LEDStateSet(BSP_LED_2, BSP_LED_STATE_OFF);
```

Parameters

Parameters	Description
led	The name of the LED to control.
state	The state to which the LED will be set: <code>BSP_LED_STATE_OFF</code> turns the LED off. <code>BSP_LED_STATE_ON</code> turns the LED on.

Function

```
void BSP_LEDStateSet ( BSP_LED led, BSP_LED_STATE state );
```

BSP_LEDToggle Function

Toggles the current state of the LED.

File

bsp_help.h

C

```
void BSP_LEDToggle(BSP_LED led);
```

Returns

None.

Description

This function toggles the current state of the LED.

- If the current state is `BSP_LED_STATE_ON` it changes to `BSP_LED_STATE_OFF`.
- If the current state is `BSP_LED_STATE_OFF` it changes to `BSP_LED_STATE_ON`.

Remarks

None.

Preconditions

`BSP_Initialize()` should have been called.

Example

```
// Toggle LED1 on the board.
BSP_LEDToggle(BSP_LED_1);

// Toggle LED2 on the board.
BSP_LEDToggle(BSP_LED_2);

// Toggle state of LED3.
BSP_LEDToggle(BSP_LED_3);
```

Parameters

Parameters	Description
led	The name of LED to toggle.

Function

```
void BSP_LEDToggle ( BSP_LED led );
```

Switch State Functions

BSP_SwitchStateGet Function

Returns the current state of the specified switch.

File

bsp_help.h

C

```
BSP_SWITCH_STATE BSP_SwitchStateGet(BSP_SWITCH bspSwitch);
```

Returns

BSP_SWITCH_STATE_PRESSED if the switch is currently pressed. BSP_SWITCH_STATE_RELEASED if the switch is not currently pressed (i.e. is released).

Description

This function returns the current state of the specified switch.

Remarks

Unless otherwise documented by the selected BSP, this function does not perform switch debouncing.

Preconditions

[BSP_Initialize\(\)](#) should have been called.

Example

```
// Check the state of the switch.
if(BSP_SWITCH_STATE_PRESSED == BSP_SwitchStateGet(BSP_SWITCH_1))
{
    // This means that Switch 1 on the board is pressed.
}
```

Parameters

Parameters	Description
switch	The name of the switch whose state to obtain.

Function

```
BSP_SWITCH_STATE BSP_SwitchStateGet ( BSP_SWITCH switch );
```

Custom Named GPIO Macros

custom_gpio_nameOff Macro

Turns off the custom-named GPIO pin.

File

bsp_help.h

C

```
#define custom_gpio_nameOff
```

Description

Custom-Named GPIO Off Macro

This is a function-like macro that turns off a custom-named GPIO pin.

Remarks

There is no actual macro or function named "custom_gpio_nameOff". It is given here as a place holder to indicate that the BSP can define custom-named macros to turn off GPIO pins with custom names for any port pins set to the "GPIO_OUT" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named GPIO off macros with names that matching the names of GPIO pins provided

on the selected board. The user can add additional custom GPIO off macros by selecting the "GPIO_OUT" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_gpio_nameOn Macro

Turns on the custom-named GPIO pin.

File

bsp_help.h

C

```
#define custom_gpio_nameOn
```

Description

Custom-Named GPIO On Macro

This is a function-like macro that turns on a custom-named GPIO pin.

Remarks

There is no actual macro or function named "custom_gpio_nameOn". It is given here as a place holder to indicate that the BSP can define custom-named macros to turn on GPIO pins with custom names for any port pins set to the "GPIO_OUT" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named GPIO on macros with names that matching the names of GPIO pins provided on the selected board. The user can add additional custom GPIO on macros by selecting the "GPIO_OUT" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_gpio_nameStateGet Macro

Gets the current value of the custom-named GPIO pin.

File

bsp_help.h

C

```
#define custom_gpio_nameStateGet
```

Description

Custom-Named GPIO Get Macro

This is a function-like macro that gets the current value of a custom-named GPIO pin.

Remarks

There is no actual macro or function named "custom_gpio_nameStateGet". It is given here as a place holder to indicate that the BSP can define custom-named macros to get the current state of GPIO pins with custom names for any port pins set to the "GPIO_IN" or "GPIO_OUT" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named GPIO state-get macros with names that matching the names of GPIO pins provided on the selected board. The user can add additional custom GPIO state-get macros by selecting the "GPIO_IN" or "GPIO_OUT" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_gpio_nameStateSet Macro

Sets a new value on the custom-named GPIO pin.

File

bsp_help.h

C

```
#define custom_gpio_nameStateSet(state)
```

Description

Custom-Named GPIO Set Macro

This is a function-like macro that sets a new value on a custom-named GPIO pin.

Remarks

There is no actual macro or function named "custom_gpio_nameStateSet". It is given here as a place holder to indicate that the BSP can define custom-named macros to set the current state of GPIO pins with custom names for any port pins set to the "GPIO_OUT" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named GPIO state-set macros with names that matching the names of GPIO pins provided on the selected board. The user can add additional custom GPIO state-set macros by selecting the "GPIO_OUT" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

Parameters

Parameters	Description
state	A boolean value. true if the pin is to be turned on. false if the pin is to be turned off.

custom_gpio_nameToggle Macro

Toggles the custom-named GPIO pin.

File

bsp_help.h

C

```
#define custom_gpio_nameToggle
```

Description

Custom-Named GPIO Toggle Macro

This is a function-like macro that toggles a custom-named GPIO pin.

Remarks

There is no actual macro or function named "custom_gpio_nameToggle". It is given here as a place holder to indicate that the BSP can define custom-named macros to toggle GPIO pins with custom names for any port pins set to the "GPIO_OUT" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named GPIO pin toggle macros with names that matching the names of GPIO pins provided on the selected board. The user can add additional custom GPIO pin toggle macros by selecting the "GPIO_OUT" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_gpio_pin_PIN Macro

Identifies the bit position within the port channel associated with a custom-named GPIO pin.

File

bsp_help.h

C

```
#define custom_gpio_pin_PIN PORTS_BIT_POS_Y
```

Description

Custom GPIO Pin Port Channel Bit Position

This macro identifies the bit position within the port channel associated with a custom-named GPIO pin. This macro is intended for use with ports system service functions.

Remarks

There is no actual macro or function named "custom_gpio_pin_PIN". It is given here as a place holder to indicate that the BSP can define custom-named macros to identify the bit position within the port channel associated with the custom-named pin.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_gpio_pin_PIN_MASK Macro

Identifies the bit mask within the port channel associated with a custom-named GPIO pin.

File

bsp_help.h

C

```
#define custom_gpio_pin_PIN_MASK (0x1 << 0)
```

Description

Custom GPIO Pin Mask

This macro identifies the bit mask within the port channel associated with a custom-named GPIO pin. This macro is intended for use with ports system service functions.

Remarks

There is no actual macro or function named "custom_gpio_pin_PIN_MASK". It is given here as a place holder to indicate that the BSP can define custom-named macros to identify the bit position within the port channel associated with the custom-named pin.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_gpio_pin_PORT Macro

Identifies the port channel associated with a custom-named GPIO pin.

File

bsp_help.h

C

```
#define custom_gpio_pin_PORT PORT_CHANNEL_x
```

Description

Custom GPIO Pin Port Channel

This macro identifies the port channel associated with a custom-named GPIO pin. This macro is intended for use with ports system service functions.

Remarks

There is no actual macro or function named "custom_gpio_pin_PORT". It is given here as a place holder to indicate that the BSP can define custom-named macros to identify the port channel associated with the custom-named pin.

This macro will actually be defined in the system_config.h header for the current configuration.

Custom Named LED Macros

custom_led_nameOff Macro

Turns off the custom-named LED.

File

bsp_help.h

C

```
#define custom_led_nameOff
```

Description

Custom-Named LED Off Macro

This is a function-like macro that turns off a custom-named LED.

Remarks

There is no actual macro or function named "custom_led_nameOff". It is given here as a place holder to indicate that the BSP can define custom-named macros to turn off LEDs with custom names for any port pins set to the "LED_AH" or "LED_AL" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named LED off macros with names that matching the names of LEDs provided on the selected board. The user can add additional custom LED off macros by selecting the "LED_AH" or "LED_AL" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_led_nameOn Macro

Turns on the custom-named LED.

File

bsp_help.h

C

```
#define custom_led_nameOn
```

Description

Custom-Named LED On Macro

This is a function-like macro that turns on a custom-named LED.

Remarks

There is no actual macro or function named "custom_led_nameOn". It is given here as a place holder to indicate that the BSP can define custom-named macros to turn on LEDs with custom names for any port pins set to the "LED_AH" or "LED_AL" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named LED on macros with names that matching the names of LEDs provided on the selected board. The user can add additional custom LED on macros by selecting the "LED_AH" or "LED_AL" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_led_nameStateGet Macro

Gets the current state of the custom-named LED.

File

bsp_help.h

C

```
#define custom_led_nameStateGet
```

Description

Custom-Named LED State Get Macro

This is a function-like macro that gets the current state of a custom named LED.

Remarks

There is no actual macro or function named "custom_led_nameStateGet". It is given here as a place holder to indicate that the BSP can define custom-named macros to get the current state LEDs with custom names for any port pins set to the "LED_AH" or "LED_AL" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named LED get-state macros with names that matching the names of LEDs provided on the selected board. The user can add additional custom LED get- state macros by selecting the "LED_AH" or "LED_AL" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

custom_led_nameToggle Macro

Toggles the custom-named LED.

File

bsp_help.h

C

```
#define custom_led_nameToggle
```

Description

Custom-Named LED Toggle Macro

This is a function-like macro that toggles a custom-named LED.

Remarks

There is no actual macro or function named "custom_led_nameToggle". It is given here as a place holder to indicate that the BSP can define custom-named macros to toggle LEDs with custom names for any port pins set to the "LED_AH" or "LED_AL" function in the Pin Settings

configuration page. By default, the selected BSP may define a number of custom-named LED toggle macros with names that matching the names of LEDs provided on the selected board. The user can add additional custom LED toggle macros by selecting the "LED_AH" or "LED_AL" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

Custom Named Switch State Macros

custom_switch_nameStateGet Macro

Gets the current value of the custom-named switch.

File

bsp_help.h

C

```
#define custom_switch_nameStateGet
```

Description

Custom-Named Switch Get Macro

This is a function-like macro that gets the current value of a custom-named switch.

Remarks

There is no actual macro or function named "custom_switch_nameStateGet". It is given here as a place holder to indicate that the BSP can define custom-named macros to get the current state of switches with custom names for any port pins set to the "SWITCH" function in the Pin Settings configuration page. By default, the selected BSP may define a number of custom-named switch state-get macros with names that matching the names of switches provided on the selected board. The user can add additional custom switch state-get macros by selecting the "SWITCH" function and defining a custom name for any desired pins in the MHC Pin Settings page.

This macro will actually be defined in the system_config.h header for the current configuration.

Data Types and Constants

BSP_LED Enumeration

Defines the names of the LEDs available on the selected board.

File

bsp_help.h

C

```
typedef enum {
    BSP_LED_1,
    BSP_LED_2,
    BSP_LED_3,
    custom_led_name
} BSP_LED;
```

Members

Members	Description
BSP_LED_1	LED 1
BSP_LED_2	LED 2
BSP_LED_3	LED 3
custom_led_name	Custom LED Name (see Remarks).

Description

BSP LED Name.

This enumeration defines the names of the LEDs available on the selected board.

Remarks

There is no actual value named "custom_led_name" in the BSP_LED enum. It is listed here as a place holder to indicate that the MHC can define custom names for any port pins set to the "LED_AH" or "LED_AL" function in the Pin Settings configuration page and those and those names will

then become part of this enum definition. By default, the selected BSP may define a number of custom LED names to match the names of LEDs provided on the selected board. The user can add additional custom switch names by selecting the "LED_AH" or "LED_AL" function and defining a custom name for any desired pins in the MHC Pin Settings page.

BSP_SWITCH_STATE Enumeration

Defines possible states of the switches on this board.

File

bsp_help.h

C

```
typedef enum {
    BSP_SWITCH_STATE_PRESSED,
    BSP_SWITCH_STATE_RELEASED
} BSP_SWITCH_STATE;
```

Members

Members	Description
BSP_SWITCH_STATE_PRESSED	Switch pressed
BSP_SWITCH_STATE_RELEASED	Switch not pressed

Description

BSP Switch state.

This enumeration defines the possible states of the switches on the selected board.

Remarks

None.

BSP_LED_STATE Enumeration

Enumerates the supported LED states.

File

bsp_help.h

C

```
typedef enum {
    BSP_LED_STATE_OFF,
    BSP_LED_STATE_ON
} BSP_LED_STATE;
```

Members

Members	Description
BSP_LED_STATE_OFF	LED State is on
BSP_LED_STATE_ON	LED State is off

Description

LED State

This enumeration defines the supported LED states.

Remarks

None.

BSP_SWITCH Enumeration

Defines the switches available on this board.

File

bsp_help.h

C

```
typedef enum {
    BSP_SWITCH_1,
    BSP_SWITCH_2,
    BSP_SWITCH_3,
    custom_switch_name
} BSP_SWITCH;
```

Members

Members	Description
BSP_SWITCH_1	SWITCH 1
BSP_SWITCH_2	SWITCH 2
BSP_SWITCH_3	SWITCH 3
custom_switch_name	Custom Switch Name (see Remarks).

Description

BSP Switch.

This enumeration defines the switches available on this board.

Remarks

There is no actual value named "custom_switch_name" in the BSP_SWITCH enum. It is listed here as a place holder to indicate that the MHC can define custom names for any port pins set to the "SWITCH" function in the Pin Settings configuration page and those and those names will then become part of this enum definition. By default, the selected BSP will define a number of custom switch names to match the names of switches provided on the selected board. The user can add additional custom switch names by selecting the "SWITCH" function and defining a custom name for any desired pins in the MHC Pin Settings page.

BSP_OSC_FREQUENCY Macro

Defines the frequency value of crystal/oscillator used on the board

File

bsp_help.h

C

```
#define BSP_OSC_FREQUENCY
```

Description

Oscillator Frequency

This macro defines the frequency value of the crystal/oscillator used on the board.

Index

A

ATWINC1500-XSTK Xplained Pro Starter Kit 4

B

BM64 Bluetooth Module Daughter Board 4

Board Support Package

bt_audio_dk 66

Development Tools 4

pic32mx_125_sk 79

pic32mx_bt_sk 81

pic32mx_eth_sk 81

pic32mx_eth_sk2 82

pic32mx_usb_sk2 83

pic32mx_usb_sk2+lcc_pictail+qvga 84

pic32mx_usb_sk2+lcc_pictail+wqvga 84

pic32mx_usb_sk2+mcb 85

pic32mx_usb_sk2+s1d_pictail+vga 85

pic32mx_usb_sk2+s1d_pictail+wqvga 86

pic32mx_usb_sk2+s1d_pictail+vwga 86

pic32mx_usb_sk2+ssd_pictail+qvga 86

pic32mx_usb_sk3 87

pic32mx270f512l_pim+bt_audio_dk 91

pic32mx460_pim+e16 93

pic32mx470_pim+e16 95

pic32mx795_pim+e16 95

pic32mz_ec_pim+bt_audio_dk 107

pic32mz_ec_pim+e16 108

pic32mz_ec_sk 109

pic32mz_ec_sk+mcb2 109

pic32mz_ec_sk+mcb2+vwga 110

pic32mz_ec_sk+s1d_pictail+wqvga 112

Board Support Packages 66

Board Support Packages Help 62

BSP_Initialize function 123

BSP_LED enumeration 132

BSP_LED_STATE enumeration 133

BSP_LEDOff function 124

BSP_LEDOn function 124

BSP_LEDStateGet function 125

BSP_LEDStateSet function 125

BSP_LEDToggle function 126

BSP_OSC_FREQUENCY macro 134

BSP_SWITCH enumeration 133

BSP_SWITCH_STATE enumeration 133

BSP_SwitchStateGet function 127

bt_audio_dk 66

bt_audio_dk+4642 67

bt_audio_dk+ak4954 68

bt_audio_dk+ak7755 69

bt_audio_dk+bm64 70

bt_audio_dk+bm64+ak4954 71

bt_sk+4642 71

bt_sk+4954 72

bt_sk+ak7755 73

C

CAN/LIN PICtail Plus Daughter Board 5

chipKIT WF32 Wi-Fi Development Board 6

chipKIT Wi-FIRE Development Board 6

chipkit_wf32 74

chipkit_wifire 75

Creating Custom Initial Board Configurations 65

custom_gpio_nameOff macro 127

custom_gpio_nameOn macro 128

custom_gpio_nameStateGet macro 128

custom_gpio_nameStateSet macro 128

custom_gpio_nameToggle macro 129

custom_gpio_pin_PIN macro 129

custom_gpio_pin_PIN_MASK macro 129

custom_gpio_pin_PORT macro 130

custom_led_nameOff macro 130

custom_led_nameOn macro 131

custom_led_nameStateGet macro 131

custom_led_nameToggle macro 131

custom_switch_nameStateGet macro 132

Customizing a BSP 63

D

Demonstration Boards

Explorer 16 Development Board 8

Graphics Controller PICtail Plus Epson S1D13517 Daughter Board 12

Graphics Display Powertip 4.3" 480x272 Board 14

Graphics Display Truly 3.2" 320x240 Board 13

Graphics Display Truly 5.7" 640x480 Board 15

Graphics Display Truly 7" 800x480 Board 16

Graphics LCD Controller PICtail Plus SSD1926 Daughter Board 17

Low-Cost Controllerless (LCC) Graphics PICtail Plus Daughter Board 20

MCP2200 Breakout Module 21

Multimedia Expansion Board (MEB) 23

PIC32 Bluetooth Audio Development Kit 31

PIC32 Bluetooth Starter Kit 32

PIC32 Ethernet Starter Kit 33

PIC32 USB Starter Kit II 36

PIC32 USB Starter Kit III 37

PIC32MX270F256D Plug-in Module (PIM) 41

PIC32MX460F512L Plug-in Module (PIM) 43

PIC32MZ Embedded Connectivity (EC) Starter Kit 49

PIC32MZ Starter Kit Adapter Board 53

PIC32MZ2048ECH100 Plug-in Module (PIM) 46

PICtail Daughter Board for SD and MMC 55

Starter Kit IO Expansion Board 58

USB PICtail Plus Daughter Board 59

dsPICDEM MCHV-3 Development Board 7

dsPICDEM MCLV-2 Development Board 7

E

Ethernet PICtail Plus Daughter Board 10

Explorer 16 Development Board 8

Explorer 16/32 Development Board 9

Extending the BSP Library Interface 63

F

Fast 100Mbps Ethernet PICtail Plus Daughter Board 11

G

Graphics Controller PICtail Plus Epson S1D13517 Daughter Board 12

Graphics Display Powertip 4.3" 480x272 Board 14

Graphics Display Truly 3.2" 320x240 Board 13

Graphics Display Truly 5.7" 640x480 Board 15

Graphics Display Truly 7" 800x480 Board 16

Graphics LCD Controller PICtail Plus SSD1926 Daughter Board 17

H

High-Performance 4.3" WQVGA Display Module with maXTouch 18

High-Performance WVGA Display Module with maXTouch 18

I

I/O1 Xplained Pro Extension Kit 19

Introduction 62

L

Library Interface 122

Low-Cost Controllerless (LCC) Graphics PICtail Plus Daughter Board 20

M

maXTouch Xplained Pro 20

MCP2200 Breakout Module 21

MikroElektronika WiFi 7 Click Board 21

Modifying the BSP Library Source Code 65

MPLAB REAL ICE 22

MRF24WG0MA Wi-Fi G PICtail/PICtail Plus Daughter Board 22

MRF24WN0MA Wi-Fi PICtail/PICtail Plus Daughter Board 23

Multimedia Expansion Board (MEB) 23

Multimedia Expansion Board II (MEB II) 25

O

OLED1 Xplained Pro Extension Kit 27

P

PIC32 Audio Codec Daughter Board - AK4642EN 29

PIC32 Audio Codec Daughter Board - AK4954A 29

PIC32 Audio Codec Daughter Board - AK7755 30

PIC32 Audio DAC Daughter Board - AK4384VT 28

PIC32 Bluetooth Audio Development Kit 31

PIC32 Bluetooth Starter Kit 32

PIC32 Ethernet Starter Kit 33

PIC32 Ethernet Starter Kit II 34

PIC32 Graphics Discovery Development Board 34

PIC32 GUI Development Board with Projected Capacitive Touch 35

PIC32 USB Digital Audio Accessory Board 36

PIC32 USB Starter Kit II 36

PIC32 USB Starter Kit III 37

pic32_gdb_ef 76

PIC32MK 100-pin Motor Control Plug-in Module (PIM) 39

PIC32MK General Purpose (GP) Development Board 38

pic32mk_gp_db 76

pic32mk_gp_db+wqvga_mxt 78

pic32mk_gp_db+wvga_mxt 79

PIC32MX XLP Starter Kit 40

pic32mx_125_sk 79

pic32mx_125_sk+lcc_pictail+qvga 80

pic32mx_270f512l_pim+ bt_audio_dk+ak4642 92

pic32mx_bt_sk 81

pic32mx_eth_sk 81

pic32mx_eth_sk2 82

pic32mx_pcap_db 82

pic32mx_usb_digital_audio_ab 82

pic32mx_usb_sk2 83

pic32mx_usb_sk2+lcc_pictail+qvga 84

pic32mx_usb_sk2+lcc_pictail+wqvga 84

pic32mx_usb_sk2+meb 85

pic32mx_usb_sk2+s1d_pictail+vga 85

pic32mx_usb_sk2+s1d_pictail+wqvga 86

pic32mx_usb_sk2+s1d_pictail+wvga 86

pic32mx_usb_sk2+ssd_pictail+qvga 86

pic32mx_usb_sk3 87

pic32mx_usb_sk3+lcc_pictail+wqvga 88

pic32mx_xlp_sk 89

PIC32MX1/2/5 Starter Kit 40

PIC32MX270F256D Plug-in Module (PIM) 41

PIC32MX270F512L Plug-in Module (PIM) 42

pic32mx270f512l_pim+bt_audio_dk 91

PIC32MX360F512L Plug-in Module (PIM) 43

PIC32MX450/470F512L Plug-in Module (PIM) 44

pic32mx460_pim+e16 93

PIC32MX460F512L Plug-in Module (PIM) 43

PIC32MX470 Curiosity Development Board 44

pic32mx470_curiosity 94

pic32mx470_pim+e16 95

PIC32MX570F512L Plug-in Module (PIM) 45

pic32mx795_pim+e16 95

PIC32MX795F512L CAN-USB Plug-in Module (PIM) 45

PIC32MZ Audio PIM 48

PIC32MZ EF Curiosity Development Board 49

PIC32MZ Embedded Connectivity (EC) Starter Kit 49

PIC32MZ Embedded Connectivity with Floating Point Unit (EF) Starter Kit 50

PIC32MZ Embedded Graphics with Disabled DRAM (DA) Starter Kit 51

PIC32MZ Embedded Graphics with External DRAM (DA) Starter Kit 51

PIC32MZ Embedded Graphics with Internal DRAM (DA) Starter Kit 52

PIC32MZ Starter Kit Adapter Board 53

pic32mz_da_sk_extddr 96

pic32mz_da_sk_extddr+meb2 97

pic32mz_da_sk_extddr+meb2+wvga 98

pic32mz_da_sk_intddr 100

pic32mz_da_sk_intddr+meb2 100

pic32mz_da_sk_intddr+meb2+wvga 101

pic32mz_da_sk_noddr+meb2 103

pic32mz_da_sk_noddr+meb2+wvga 105

pic32mz_ec_pim+bt_audio_dk 107

pic32mz_ec_pim+e16 108

pic32mz_ec_sk 109

pic32mz_ec_sk+meb2 109

pic32mz_ec_sk+meb2+wvga 110

pic32mz_ec_sk+s1d_pictail+vga 111

pic32mz_ec_sk+s1d_pictail+wqvga 112

pic32mz_ec_sk+s1d_pictail+wvga 113

pic32mz_ef_curiosity 113

pic32mz_ef_pim+bt_audio_dk 114
pic32mz_ef_pim+bt_audio_dk+ak4642 115
pic32mz_ef_pim+e16 116
pic32mz_ef_sk 117
pic32mz_ef_sk+maxtouch_xplained_pro_3_5 118
pic32mz_ef_sk+meb2 118
pic32mz_ef_sk+meb2+wvga 119
pic32mz_ef_sk+s1d_pictail+vga 120
pic32mz_ef_sk+s1d_pictail+wqvga 121
PIC32MZ2048ECH100 Plug-in Module (PIM) 46
PIC32MZ2048ECH144 Audio Plug-in Module (PIM) 47
PIC32MZ2048EFH100 Plug-in Module (PIM) 46
PIC32MZ2048EFH144 Audio Plug-in Module (PIM) 47
PIC32WK Wi-Fi Starter Kit 53
pic32wk_gbp_gpd_sk+module 122
PICkit 3 In-Circuit Debugger 54
PICKtail Daughter Board for SD and MMC 55
PICKtail Plus Expansion Board 56
Prototype PICKtail Plus Daughter Board 57

S

SAM V71 Xplained Ultra Evaluation Kit 58
Starter Kit I/O Expansion Board 58
Supported Development Boards 4
Supported Device Families 3

U

USB PICKtail Plus Daughter Board 59
Using a BSP 62

V

Volume II: Supported Hardware 2

W

WILC1000 Wi-Fi PICKtail/PICKtail Plus Daughter Board 60
WINC1500 Wi-Fi module 60
WINC1500 Wi-Fi PICKtail/PICKtail Plus Daughter Board 61