PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit

EV16B95A

Product Page Links

Preface

The PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit is a hardware platform which uses the PIC32CM6408JH00064 microcontroller units (MCUs). The evaluation kit provides easy access to the microcontroller's features and may be used to develop custom applications.

The evaluation kit comes pre-programmed with a stand-alone demonstration application and is powered through its USB-C connection. The evaluation kit can be used as a stand-alone discovery element and it can be combined with expansion elements for quick prototyping.

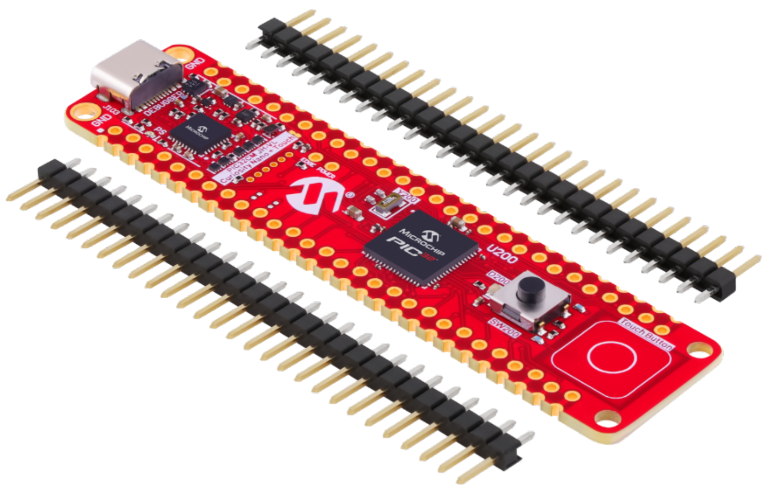


Table of Contents

Preface........ 1

1. Introduction........ 3
    1.1. Processor Overview........ 3
    1.2. Board Overview........ 3
    1.3. Board Features........ 3
    1.4. Total System Solutions (TSS)........ 4

2. Getting Started........ 5
    2.1. Curiosity Nano+ Touch Quick Start........ 5
    2.2. How the Curiosity Nano Fits Into the MPLAB Tools Ecosystem........ 5
    2.3. MPLAB Data Visualizer Support for Curiosity Nano........ 5
    2.4. Using Pin-Headers........ 5

3. On-Board Debugger........ 7
    3.1. On-Board Debugger Overview........ 7
    3.2. On-Board Debugger Connections........ 7
    3.3. Debugger USB Enumeration........ 8
    3.4. Virtual Serial Port (CDC)........ 8
    3.5. Data Gateway Interface (DGI)........ 11
    3.6. Disconnecting the On-Board Debugger........ 12

4. Hardware........ 13
    4.1. Pinout........ 13
    4.2. Crystal........ 14
    4.3. LED........ 14
    4.4. Mechanical Switch........ 14
    4.5. Touch Button........ 14
    4.6. Power Supply........ 15

5. Appendix........ 19
    5.1. Schematics........ 19
    5.2. Assembly Drawing........ 22
    5.3. Bill of Materials........ 22

6. Revision History........ 24

Microchip Information........ 25
    Trademarks........ 25
    Legal Notice........ 25
    Microchip Devices Code Protection Feature........ 25

Product Page Links........ 26

1. Introduction

1.1. Processor Overview

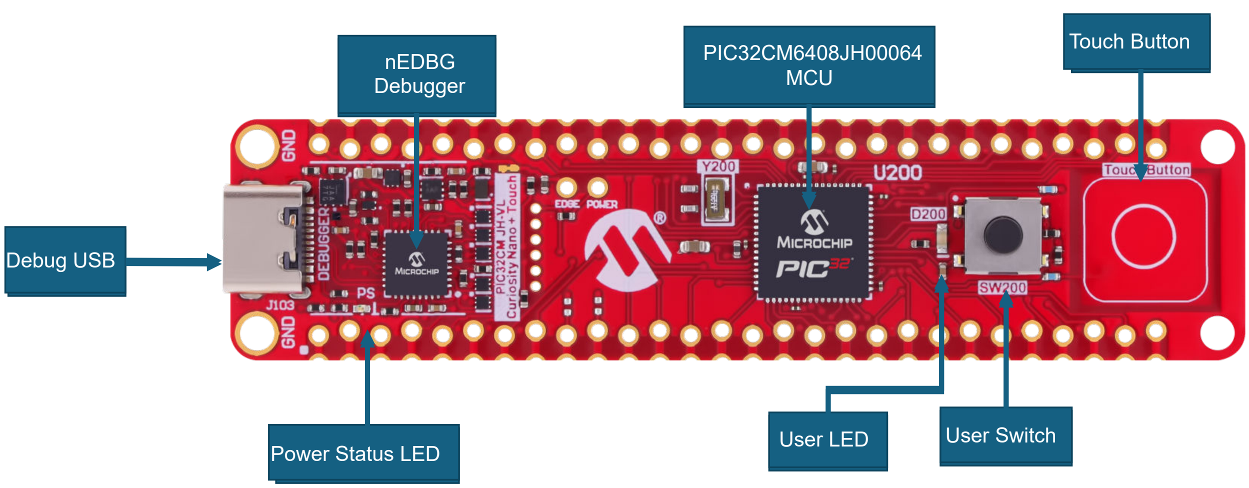
The following are the key features of the PIC32CM6408JH00064 MCU:

- For performance
  - Arm® Cortex®-M0+ CPU
    - Running at up to 48 MHz with 64 Kb Flash and 8 Kb SRAM
    - Single-cycle hardware multiplier
    - Micro Trace Buffer
    - Memory Protection Unit (MPU)
  - For low power
    - Idle and Standby Sleep modes
    - SleepWalking peripherals
  - For touch input
    - 256-Channel capacitive touch and proximity sensing
    - Low-power, high-sensitivity, environmentally robust capacitive touch buttons, sliders, and wheels

1.2. Board Overview

The important sections of the PIC32CM JH-Value Line Curiosity Nano+ Touch board are highlighted in the following figure.

Figure 1-1. PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit Overview



1.3. Board Features

The PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit has the following key features:

- The PIC32CM6408JH00064 MCU
- User application yellow LED
- User application switch
- User application touch button
- USB-C for debugger

- Can be used for powering the board
- Must be used to program or debug the board
- On-board nano debugger (nEDBG)
  - Board identification in MPLAB® X IDE
  - One green power/status LED
  - Programming and debugging
  - Communications Device Class (CDC) virtual COM port
  - One logic analyzer DGI GPIO
  - The target device is programmed and debugged by the on-board Nano debugger, therefore no external programmer or debugging tool is required
- Adjustable target voltage
  - MIC5353 LDO regulator controlled by the on-board debugger
  - 1.7V to 3.6V output voltage
  - 500 mA maximum output current (limited by ambient temperature and output voltage)

1.4. Total System Solutions (TSS)

The table below shows the list of components part of the Total System Solutions (TSS) on the PIC32CM JH-Value Line Curiosity Nano+ Touch board.

Table 1-1. PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit Microchip Total System Solutions (TSS)

TSS Component	Qty (per board)	Function
MIC5353YMT	1	ANALOG ADJUSTABLE LDO 500 mA
MIC94163YCS	1	IC LOAD SWITCH 3A
MIC5528-3.3YMT	1	ANALOG LDO 3.3V
MIC2008YML	1	ANALOG POWER SWITCH 5.5V 2.1A
ATSAMD21E18A-MUT	1	nEDBG MICROCONTROLLER
PIC32CM6408JH00064-I/5LX	1	TARGET MICROCONTROLLER
VMK3-9001-32K7680000TR	1	CRYSTAL 32.768 kHz

2. Getting Started

2.1. Curiosity Nano+ Touch Quick Start

Users need to follow these steps to explore the Curiosity Nano+ Touch platform:

1. Download MPLAB X IDE.
2. Launch MPLAB X IDE.
3. Connect a USB cable (Standard-A to USB-C) between the PC and the debug USB port on the kit.

2.2. How the Curiosity Nano Fits Into the MPLAB Tools Ecosystem

When the board connects to the computer for the first time, the operating system will install the driver software. The drivers for the board are included with MPLAB X IDE. Once this is done, when connecting the Curiosity Nano to a host PC via USB, if MPLAB X IDE is open, a Kit Window is opened with several key links for that Curiosity Nano. When creating a new project, the part number on the Curiosity Nano will be detected, as will the debug tool.

2.3. MPLAB Data Visualizer Support for Curiosity Nano

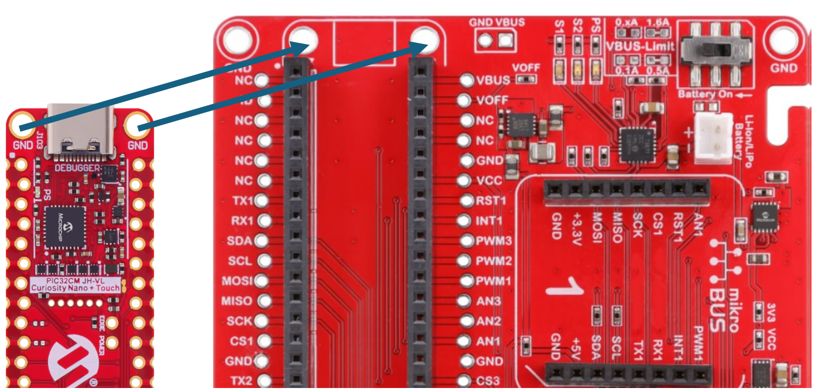
The Curiosity Nano, via a USB/serial bridge, facilitates a connection between a UART on the Target MCU and the computer's COM port. For example, the user may use this to connect to the MPLAB Data Visualizer or other terminal programs.

2.4. Using Pin-Headers

The edge connector footprint on the PIC32CM JH-Value Line Curiosity Nano+ Touch board has a staggered design where each hole is shifted 8 mils (~0.2 mm) off-center. The hole shift allows the use of the regular 100 mil pin headers without soldering on the board. The pin-headers can be used in applications, such as pin sockets and prototyping boards without issues once both are firmly in place.

Figure 2-1. Attaching Pin-Headers to the Curiosity Nano Board

Figure 2-2. Connecting to Curiosity Nano Base for Click board™



Tip:

- Start at one end of the pin-header and gradually insert the header along the length of the board. Once all the pins are in place, use a flat surface to push them in.
- For applications using the pin-headers permanently, it is still recommended to solder them in place.
- Once the pin-headers are in place, are hard to remove by hand. Use a set of pliers and carefully remove the pin-headers to avoid damage the pin-headers and PCB.

3. On-Board Debugger

3.1. On-Board Debugger Overview

The PIC32CM6408JH00064 Curiosity Nano contains an on-board debugger for programming and debugging. The on-board debugger is a composite USB device consisting of several interfaces:

- A debugger that can program and debug the PIC32CM6408JH00064 in MPLAB X IDE.
- A virtual serial port (CDC) that is connected to a Universal Asynchronous Receiver/Transmitter (UART) on the PIC32CM6408JH00064 and provides an easy way to communicate with the target application through terminal software.
- A Data Gateway Interface (DGI) for code instrumentation with logic analyzer channels (debug GPIO) to visualize program flow.

The on-board debugger controls a Power and Status LED (marked as PS) on the PIC32CM JH-Value Line Curiosity Nano+ Touch board. The table below shows how the different operation modes control the LED.

Table 3-1. On-Board Debugger LED Control

Operation Mode	Status LED
Boot Loader mode	LED blinks slowly at 1 Hz during power up.
Power-up	The LED is ON.
Normal operation	The LED is ON.
Programming	Activity indicator, the LED flashes slowly during programming or debugging.
Fault	The LED flashes fast if a power fault is detected.
Sleep/Off	LED is OFF. The on-board debugger is either in sleep-mode or power-down mode. This will occur only if the kit is externally powered.

Info: Slow blinking is approximately 1 Hz and rapid blinking is about 5 Hz.

3.2. On-Board Debugger Connections

The table below shows the connections between the target and the debugger. All the connections between the target and the debugger are tri-stated when the debugger is not using the interface. Therefore, there are few contaminations of the signals, for example, the pins can be configured to anything the user wants.

Info: The 12-edge connections closest to the USB connector on Curiosity Nano boards have a standardized pinout. The program/debug pins have different functions depending on the target programming interface.

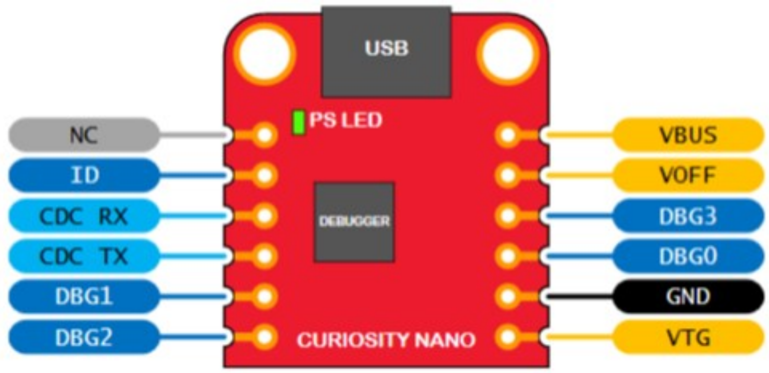
Table 3-2. On-Board Debugger Connections

Debugger Pin	PIC32CM6408JH00064 Pin		Description
CDC TX	PA17	USART1 RX	USB CDC TX line
CDC RX	PA16	USART1 TX	USB CDC RX line
DBG0	PA31	SWDIO	Debug data line
DBG1	PA30	SWCLK	Debug clock line
DBG2	PB31	GPIO	Debug GPIO0/SW0
DBG3	nRESET	RESET	Reset line
ID			ID line for extensions
NC			No connect
V_{BUS}			VBUS voltage for external use.

Table 3-2. On-Board Debugger Connections (continued)

Debugger Pin	PIC32CM6408JH00064 Pin		Description
nVOFF			Voltage Off input. Disables the target regulator and target voltage when pulled low.
VTG			Target voltage
GND			Common ground

Figure 3-1. Curiosity Nano Debugger Pinout



3.3. Debugger USB Enumeration

The on-board debugger on the PIC32CM JH-Value Line Curiosity Nano+ Touch board appears as a Human Interface Device (HID) on the host computer's USB subsystem. The debugger supports full-featured programming and debugging of the PIC32CM6408JH00064 using MPLAB X IDE and some third-party IDEs.

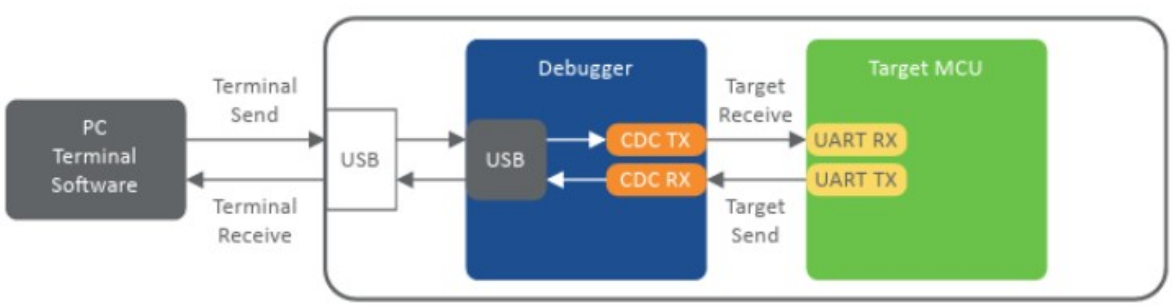
3.4. Virtual Serial Port (CDC)

A virtual communications port is provided by a general-purpose USB serial bridge between the host PC and the target device.

3.4.1. Overview

The on-board debugger implements a composite USB device with a standard Communications Device Class (CDC) interface, which appears on the host as a virtual serial port. Use the CDC to stream arbitrary data between the host computer and the target in both directions: All characters sent through the virtual serial port on the host computer will be transmitted as UART on the debugger's CDC TX pin. The UART characters captured on the debugger's CDC RX pin will be returned to the host computer through the virtual serial port.

Figure 3-2. CDC Connection



Info: The debugger's CDC TX pin is connected to a UART RX pin on the target for receiving characters from the host computer, as shown in the figure above. Similarly, the debugger's CDC RX pin is connected to a UART TX pin on the target for transmitting characters to the host computer.

3.4.2. Operating System Support

On Windows® machines, the CDC will enumerate as Curiosity Virtual COM Port and appear in the Ports section of the Windows Device Manager. The COM port number can be found there.

Info: On older Windows systems, the CDC requires a USB driver. The MPLAB X IDE installation includes this driver.

On Linux® machines, the CDC will enumerate and appear as /dev/ttyACM#.

Info: tty* devices belong to the "dialout" group in Linux, so it may be necessary to become a member of that group to have permission to access the CDC.

On Mac® machines, the CDC will enumerate and appear as /dev/tty.usbmodem#. Depending on the terminal program used, it will appear in the available list of modems as usbmodem#.

Info: For all operating systems, use a terminal emulator that supports DTR signaling. See Signaling section.

3.4.3. Limitations

Not all UART features are implemented in the on-board debugger CDC. The constraints are outlined here:

- Baud rate: Must be in the range of 1200 bps to 500 kbps. Any baud rate outside this range will be set to the closest limit without warning. Baud rate can be changed on-the-fly.
- Character format: Only 8-bit characters are supported.
- Parity: Can be odd, even, or none.
- Hardware flow control: Not supported.
- Stop bits: One or two bits are supported.

3.4.4. Signaling

During USB enumeration, the host OS will start the communication and data pipes of the CDC interface. At this point, it is possible to set and read back the baud rate and other UART parameters of the CDC, but sending and receiving data will not be enabled.

The terminal must assert the DTR signal when it connects to the host. As this is a virtual control signal implemented on the USB interface, it is not physically present on the board. Asserting the DTR signal from the host will indicate to the on-board debugger that a CDC session is active. The debugger will enable its level shifters (if available) and start the CDC data send and receive mechanisms.

Deasserting DTR in debugger firmware version 1.20 or earlier has the following behavior:

- Debugger UART receiver is disabled, and no further data will be transferred to the host computer.
- Debugger UART transmitter will continue to send queued data ready for transfer, but no new data is accepted from the host computer.
- Level shifters (if available) are not disabled, and the debugger CDC TX line remains driven.

Deasserting DTR in debugger firmware version 1.21 or later has the following behavior:

- Debugger UART receiver is disabled, and no further data will be transferred to the host computer.
- Debugger UART transmitter will continue to send queued data ready for transfer, but no new data is accepted from the host computer.

- Once the ongoing transmission is complete, level shifters (if available) are disabled, and the debugger CDC TX line will become high impedance.

Remember: Set up the terminal emulator to assert the DTR signal. Without the signal, the on-board debugger will not send or receive data through its UART.

Tip: The on-board debugger's CDC TX pin will not be driven until the CDC interface is enabled by the host computer. There are no external pull-up resistors on the CDC lines connecting the debugger and the target, meaning the lines are floating during power-up. The target device may enable the internal pull-up resistor on the pin that is connected to the debugger's CDC TX pin to avoid glitches resulting in unpredictable behavior, such as framing errors.

3.4.5. Advanced Use

CDC Override Mode

In ordinary operation, the on-board debugger is a true UART bridge between the host and the device. However, in certain use cases, the on-board debugger can override the basic Operating mode and use the CDC TX and RX pins for other purposes.

Dropping a text file into the on-board debugger's mass storage drive can send characters out of the debugger's CDC TX pin. The file name and extension are trivial, but the text file will start with the characters:

```
CMD:SEND_UART=
```

Debugger firmware version 1.20 or earlier has the following limitations:

- The maximum message length is 50 characters – all remaining data in the frame are ignored.
- The default baud rate used in this mode is 9600 bps, but if the CDC is already active or configured, the previously used baud rate still applies.

Debugger firmware version 1.21 and later has the following limitations/features:

- The maximum message length will vary depending on the MSC/SCSI layer timeouts on the host computer and/or operating system. A single SCSI frame of 512 bytes (498 characters of payload) is ensured, and files up to 4 KB will work on most systems. The transfer will be completed on the first NULL character encountered in the file.
- The baud rate used is always 9600 bps for the default command:
  `CMD:SEND_UART=`
- Do not use the CDC Override mode simultaneously with data transfer over the CDC/terminal. If a CDC terminal session is active when receiving a file via the CDC Override mode, it will be suspended for the duration of the operation and resumed once complete.
- Additional commands are supported with explicit baud rates:
  `CMD:SEND_9600=`
  `CMD:SEND_115200=`
  `CMD:SEND_460800=`

USB-Level Framing Considerations

Sending data from the host to the CDC can be done byte-wise or in blocks, chunked into 64-byte USB frames. Each frame will be queued for transfer to the debugger's CDC TX pin. Sending a small amount of data per frame can be inefficient, particularly at low baud rates, as the on-board debugger buffers frames but not bytes. A maximum of four 64-byte frames can be active at any time. The on-board debugger will throttle the incoming frames accordingly. Sending full 64-byte frames containing data is the most efficient method.

When receiving data on the debugger's CDC RX pin, the on-board debugger will queue up the incoming bytes into 64-byte frames, which are sent to the USB queue for transmission to the host when are full. Incomplete frames are pushed to the USB queue at approximately 100 ms intervals, triggered by USB start-of-frame tokens. Up to eight 64-byte frames can be active at any time.

An overrun will occur if the host (or the software running) fails to receive data fast enough. When this happens, the last-filled buffer frame recycles instead of being sent to the USB queue, and a complete data frame will be lost. To prevent this occurrence, the user will ensure that the CDC data pipe is continuously read, or the incoming data rate will be reduced.

Sending Break Characters

The host can send a UART break character to the device using the CDC, which can be useable for resetting a receiver state-machine or signaling an exception condition from the host to the application running on the device.

A break character is a sequence of at least 11 zero bits transmitted from the host to the device.

Not all UART receivers have support for detecting a break, but a correctly-formed break character usually triggers a framing error on the receiver.

Sending a break character using the debugger's CDC has the following limitations:

- Sending a break will cause any data being sent to be lost. Be sure to wait a sufficient amount of time to allow all characters in the transmission buffer to be sent (see above section) before sending the break, which is in line with expected break character usage. For example, reset a receiver state-machine after a timeout occurs waiting for returning data to the host.
- The CDC specification allows for debugger-timed breaks of up to 65534 ms in duration to be requested. For simplicity, the debugger will limit the break duration to a maximum of 11 bit durations at its minimum supported baud rate.
- The CDC specification allows for indefinite host-timed breaks. It is the terminal application/user's responsibility to release the break state in this case.

Note: Sending break characters is available in debugger firmware version 1.24 and later.

3.5. Data Gateway Interface (DGI)

Data Gateway Interface (DGI) is a USB interface transporting raw and timestamped data between on-board debuggers and host computer-based visualization tools. MPLAB Data Visualizer is used on the host computer to display any debug GPIO data. It is available as a plug-in for MPLAB X IDE or a stand-alone application that can be used in parallel with MPLAB X IDE.

Although DGI encompasses several physical data interfaces, the PIC32CM JH-Value Line Curiosity Nano+ Touch implementation includes logic analyzer channels: Two debug GPIO channels (known as DGI GPIO).

3.5.1. Debug GPIO

Debug GPIO channels are timestamped digital signal lines connecting the target application to a host computer visualization application. They are typically used to plot low-frequency events on a time axis, such as when given Application state transitions occur.

Debug GPIO channels are timestamped, so the resolution of DGI GPIO events is determined by the resolution of the DGI Timestamp module.

3.5.2. Timestamping

When captured by the debugger, DGI sources are timestamped. The timestamp counter implemented in the Curiosity Nano debugger increments at a 2 MHz frequency, providing a timestamp resolution of a half microsecond.

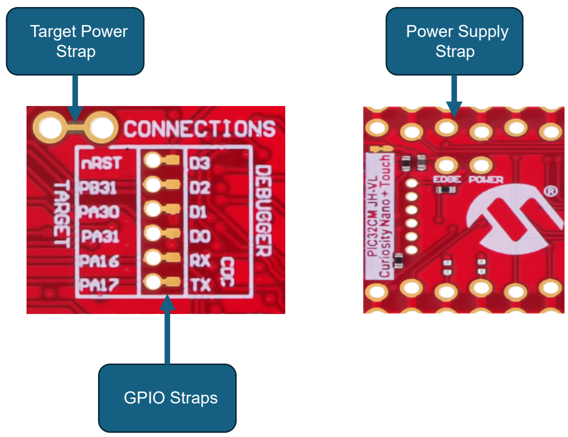
3.6. Disconnecting the On-Board Debugger

The on-board debugger and level shifters can be disconnected from the PIC32CM6408JH00064 microcontroller. The power supply block diagram (Figure 4-4) shows all connections between the debugger and the PIC32CM6408JH00064 microcontroller. The signal names are printed in silkscreen on the bottom side of the board. To disconnect the debugger, cut the GPIO straps as shown in Figure 3-3.

Notes:

- Cutting the connections to the debugger will disable programming, debugging, and data streaming. The signals will be disconnected from the board edge next to the on-board debugger section.
- Solder 0Ω resistors across the footprints, or short-circuit traces with tin solder, to reconnect any cut signals.

Figure 3-3. Common Curiosity Nano Cut Straps



4. Hardware

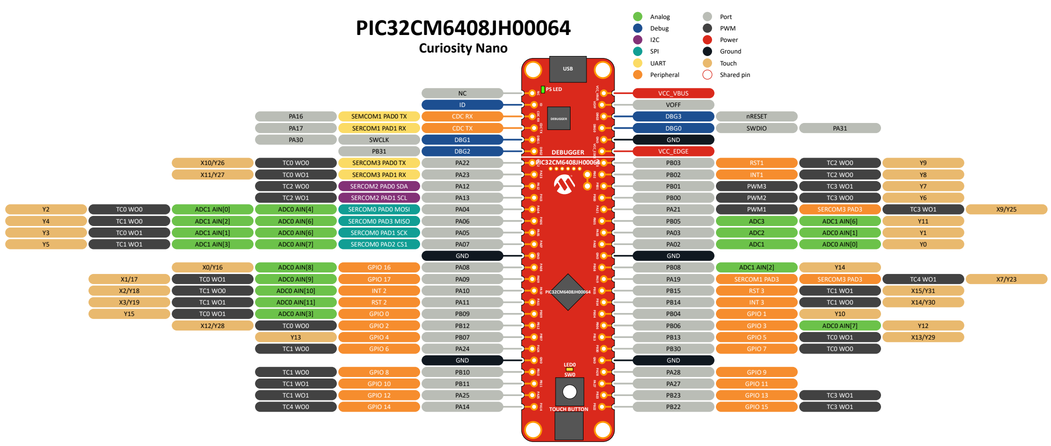
4.1. Pinout

4.1.1. PIC32CM JH-Value Line Curiosity Nano Pinout

The I/O pins of the PIC32CM6408JH00064 MCU are accessible through the edge connectors on the board. The image below shows the board pinout.

Refer to the *64-pin VQFN/64-pin TQFP Pinout and Multiplexing* section in the "PIC32CM64/32 JH00 Family Data Sheet" (*DS60001880*) for all available functions on each pin.

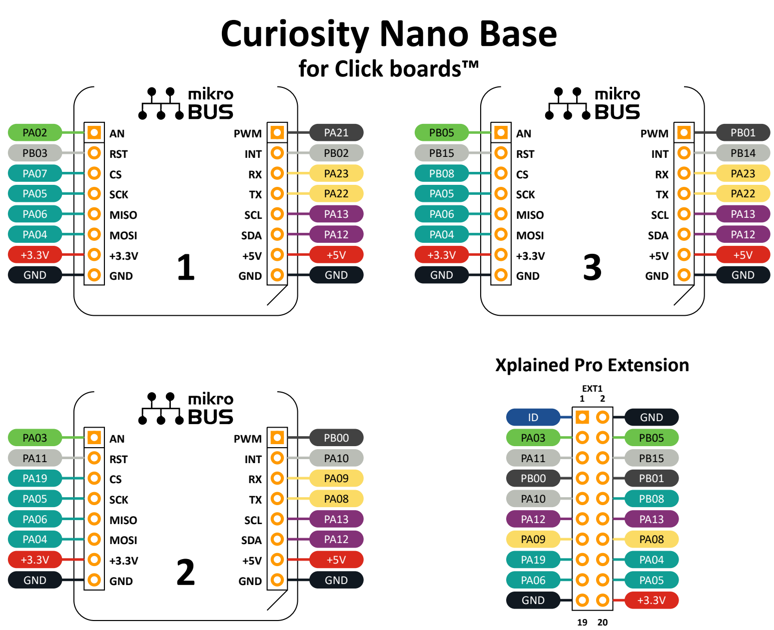
Figure 4-1. PIC32CM JH-Value Line Curiosity Nano+ Touch Pinout



4.1.2. Pinouts Compatibility with the Expansion Base Boards

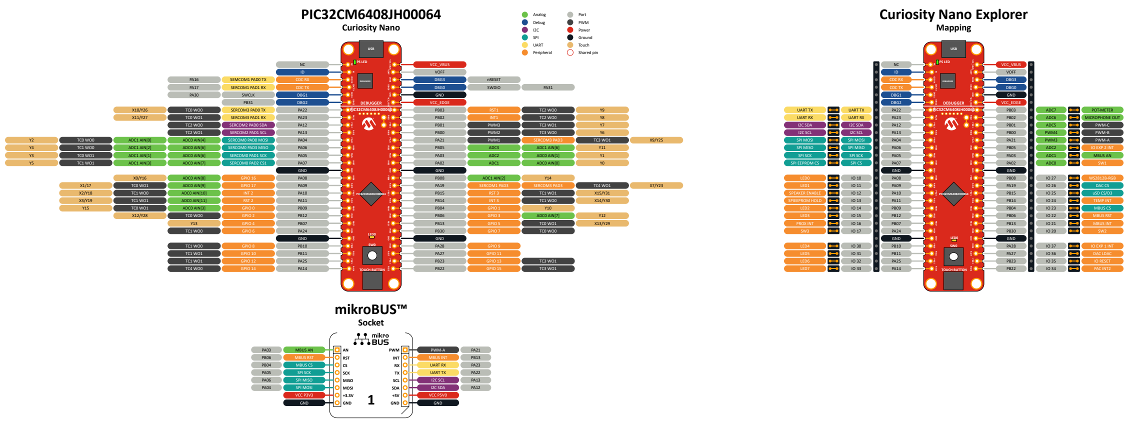
The following figure shows a simplified reference to enable quicker development with the Curiosity Nano Base for Click Boards™ (AC164162).

Figure 4-2. PIC32CM JH-Value Line Curiosity Nano+ Touch Connections to the Curiosity Nano Base for Click board



The following figure shows a simplified reference to enable quicker development with the Curiosity Nano Explorer board (EV58G97A).

Figure 4-3. PIC32CM JH-Value Line Curiosity Nano+ Touch Connections to the Curiosity Nano Explorer



4.2. Crystal

The PIC32CM JH-Value Line Curiosity Nano+ Touch board has a 32.768 kHz crystal mounted. The crystal is connected to the PIC32CM6408JH00064 microcontroller by default.

Table 4-1. Crystal Connections

PIC32CM6408JH00064 Pin	Function
PA00	XIN32
PA01	XOUT32

4.3. LED

One yellow LED is available on the PIC32CM JH-Value Line Curiosity Nano+ Touch board. The LED is intended to be used by the user's application and is controlled as a GPIO.

Table 4-2. LED Connection

PIC32CM6408JH00064 Pin	Function
PB17	Yellow LED

4.4. Mechanical Switch

The PIC32CM JH-Value Line Curiosity Nano+ Touch board has one mechanical switch, a generic user-configurable switch. Pressing the mechanical switch will connect the I/O pin to the ground (GND).

Table 4-3. Mechanical Switch

PIC32CM6408JH00064 Pin	Function
PB16	Mechanical Switch

4.5. Touch Button

The PIC32CM JH-Value Line Curiosity Nano+ Touch board has one QTouch button with a driven shield. The QTouch button is implemented using the built-in Peripheral Touch Controller (PTC) of the MCU.

Note: The touch button on the board is to showcase touch capability only and is not intended for performance evaluation. To evaluate the performance, use the Curiosity Nano Touch Adapter board (AC80T88A) with one of the Touch Extension boards or design the PCBs by referring to the "Capacitive Touch Sensor Design Guide" (*DS00002934*).

Table 4-4. Touch Button

PIC32CM6408JH00064 Pin	Function
PA18	QTouch Button
PA20	QTouch Button Shield

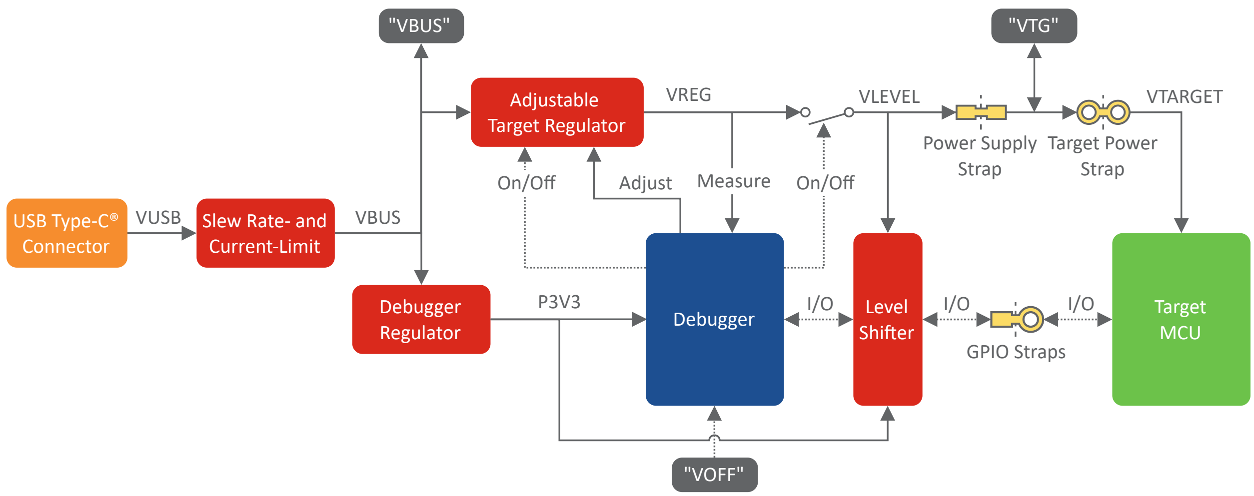
4.6. Power Supply

The USB-C port powers the board. The VBUS net is limited to a 2 V/ms slew rate and is current limited to 500 mA by U108 (MIC2008).

The power supply consists of two LDO regulators, one to generate 3.3V for the on-board debugger and an adjustable LDO regulator for the target PIC32CM6408JH00064 microcontroller and its peripherals. The voltage from a USB connector can vary between 4.4V and 5.25V (according to the USB specification) and will limit the maximum voltage supplied to the target.

The figure below shows the entire power supply system on the PIC32CM JH-Value Line Curiosity Nano+ Touch board.

Figure 4-4. Power Supply Block Diagram



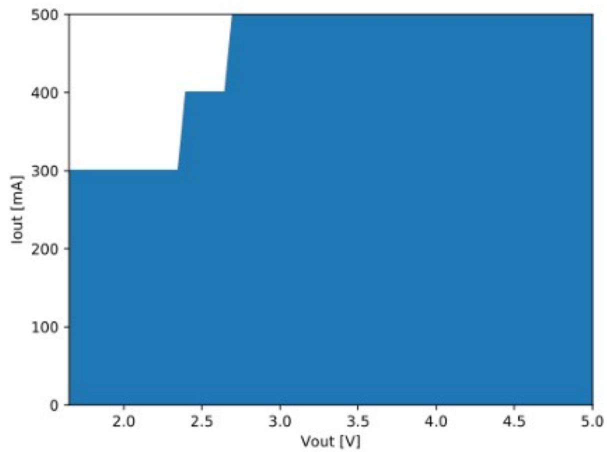
4.6.1. Target Regulator

The target voltage regulator is a MIC5353 variable output LDO. The on-board debugger can adjust the voltage output supplied to the board target section by manipulating the MIC5353's feedback voltage. The hardware implementation is limited to an approximate voltage range from 1.7V to 5.1V. Additional output voltage limits are configured in the debugger firmware to ensure that the output voltage never exceeds the hardware limits of the PIC32CM6408JH00064 microcontroller. The voltage limits configured in the on-board debugger on PIC32CM JH-Value Line Curiosity Nano+ Touch board are 1.8 – 5.5V.

Info: The factory default target voltage is 5.0V. Any change to the target voltage is persistent, even after a power toggle. The resolution is less than 5 mV but may be limited to 10 mV by the adjustment program.

Info: The voltage settings setup in MPLAB X IDE is not applied immediately to the board. Like clicking the Refresh Debug Tool Status button in the project dashboard tab or programming/reading the program memory, the new voltage setting is applied to the board when accessing the debugger.

Figure 4-5. Target Regulator Safe Operation Area



The voltage output of the target regulator is continuously monitored (measured) by the on-board debugger. An error condition will be flagged and the target voltage regulator will be switched off, detecting and handling any short-circuit conditions if it is more than 100 mV over/under the set device voltage. It will detect and handle if an external voltage, which causes VCC_TARGET to move outside the voltage setting monitoring window of ±100 mV, is suddenly applied to the VTG pin without setting the VOFF pin low.

Info: The on-board debugger has a monitoring window of VCC_TARGET±100 mV, and the status LED will blink rapidly if the external voltage is under this limit. The on-board debugger status LED will continue to shine if the external voltage surpasses this limit. When removing the external voltage, the status LED will start blinking rapidly until the on-board debugger detects the new situation and turns the target voltage regulator back on.

4.6.2. External Supply

Instead of the on-board target regulator, an external voltage can power the PIC32CM JH-Value Line Curiosity Nano+ Touch board. When shorting the Voltage Off (VOFF) pin to the ground (GND) pin, the on-board debugger firmware disables the target regulator, and it is safe to apply an external voltage to the VTG pin.

It is safe to apply an external voltage to the VTG pin when no USB cable is plugged into the DEBUG connector on the board. The VOFF pin can be tied low/let go at any time, which will be detected by a pin-change interrupt to the on-board debugger, which controls the target voltage regulator accordingly.

WARNING Applying an external voltage to the VTG pin without shorting VOFF to GND may cause permanent damage to the kit.

WARNING Absolute maximum external voltage is 5.5V for the level shifters on board. Applying a higher voltage may cause permanent damage to the kit.

WARNING Do not apply any voltage to the VOFF pin. Let the pin float to enable the power supply.

Programming, debugging, and data streaming are still possible with an external power supply. The USB cable will power the debugger and signal level shifters. Both regulators, the debugger, and the level shifters are powered down when the USB cable is removed.

Info: The on-board debugger monitors the voltage supplied to the board. If VOFF is not pulled low and the external power supplied differs by more than ±100 mV from the target regulator setting, the on-board debugger will shut off the target regulator and begin blinking the status LED rapidly, indicating an error condition. Once the input voltage returns within ±100 mV of the target regulator setting, the on-board debugger will switch on the target regulator and stop blinking the status LED.

Info: In addition to the power consumed by the PIC32CM6408JH00064 microcontroller and its peripherals, approximately 100 µA will be drawn from any external power source to power the on-board level shifters and voltage monitor circuitry when plugging a USB cable into the DEBUG connector on the board. When a USB cable is unplugged, some current is used to supply the level shifter's voltage pins, having a worst-case current consumption of approximately 5 µA. Typical values may be as low as 100 nA.

4.6.3. Power Supply Exceptions

This section sums up most exceptions that can occur with the power supply.

Target Voltage Shuts Down

Not reaching the target voltage setting can happen if the target section draws too much current at a given voltage and causes the thermal shutdown safety feature of the MIC5353 regulator to kick in. To avoid this, reduce the current load of the target section.

Target Voltage Setting is Not Reached

The USB input voltage (specified to be 4.4-5.25V) limits the maximum output voltage of the MIC5353 regulator at a given voltage setting and current consumption. If a higher output voltage is needed, use a USB power source with a higher input voltage or use an external voltage supply on the VTG pin.

Target Voltage is Different From Setting

An externally applied voltage to the VTG pin without setting the VOFF pin low can cause this. If the target voltage varies more than 100 mV over/under the voltage setting, the on-board debugger will detect it, and the internal voltage regulator will shut down. To fix this issue, remove the applied voltage from the VTG pin, and the on-board debugger will enable the on-board voltage regulator when the new condition is detected. Note that the PS LED will blink rapidly if the target voltage is below 100 mV of the setting but will ordinarily turn on when it is more than 100 mV above it.

No, or Very Low Target Voltage and PS LED is Blinking Rapidly

A full or partial short circuit can cause this and is a particular case of the issue above. Remove it, and the on-board debugger will re-enable the on-board target voltage regulator.

No Target Voltage and PS LED is Lit 1

This situation occurs if the target voltage is set to 0.0V. Set the target voltage to a value within the specified voltage range for the target device to fix this.

No Target Voltage and PS LED is Lit 2

This situation can be the issue when cutting power jumper J200 and/or J201 and setting the target voltage regulator to a value within the specified voltage range for the target device. To fix this, solder a wire/bridge between the pads for J200/J201 or add a jumper on J201 if a pin-header is mounted.

V_{BUS} Output Voltage is Low or Not Present

If the VBUS output voltage is low or missing, the reason is probably a high-current drain on VBUS, and the current limit set by U202 (MIC2008) is tripped and has cut off VBUS completely. Reduce the current consumption on the VBUS pin to fix this issue.

4.6.4. Low-Power Measurement

Power to the PIC32CM6408JH00064 microcontroller comes from the on-board power supply and VTG pin through a 100-mil pin-header marked with POWER in silkscreen (J101). To measure the power consumption of the PIC32CM6408JH00064 microcontroller and other peripherals connected to the board, cut the Target Power strap (J101) on the bottom side and connect an ammeter across it.

Tip: A 100-mil pin-header can be soldered into the Target Power strap (J101) footprint for a simple connection of an ammeter. Place a jumper cap on the pin-header once the ammeter is no longer needed.

To measure the lowest possible power consumption, follow these steps:

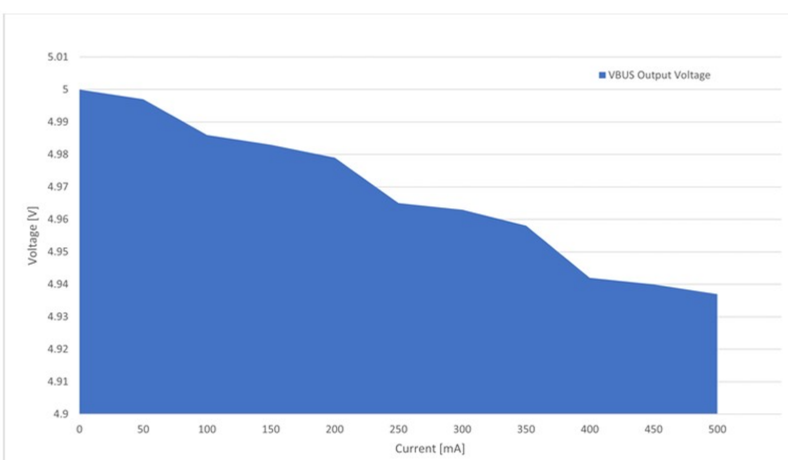
1. Cut the POWER strap with a sharp tool.
2. Solder a 1x2 100 mil pin-header in the footprint.
3. Connect an ammeter to the pin header.
4. Write firmware that:
   a. Tri-states any I/O connected to the on-board debugger.
   b. Sets the microcontroller in its lowest Power sleep mode.
5. Program the firmware into the PIC32CM6408JH00064 microcontroller.

Info: The on-board level shifters will draw a small amount of current even when unused. Each level shifter has a maximum of 2 µA leakage current. Therefore, the worst-case maximum current draw for the five on-board level shifters is 10 µA. Prevent leakage current through an I/O pin connected to a level shifter by keeping the I/O pin tri-stated. All I/Os connected to the on-board debugger are listed in On-Board Debugger Connections section. The on-board level shifters can be completely disconnected, preventing leakage, as described in Disconnecting the On-Board Debugger section.

4.6.5. VBUS Output Pin

The PIC32CM JH-Value Line Curiosity Nano+ Touch board has a VBUS output pin that can be used to power external components that need a 5V supply. The VBUS output pin is protected by the same start-up delay with a slew rate and current limiter as the rest of the power supply. A side effect is a voltage drop on the VBUS output with higher current loads. The chart below shows the VBUS output voltage versus the current load of the VBUS output.

Figure 4-6. VBUS Output Voltage vs. Current



5. Appendix

5.1. Schematics

This section contains the following schematics for the PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit:

- Kit Overview Schematic, Figure 5-1
- Power Supply and Debugger Schematic, Figure 5-2
- Target Microcontroller Schematic, Figure 5-3

Figure 5-1. Kit Overview Schematic

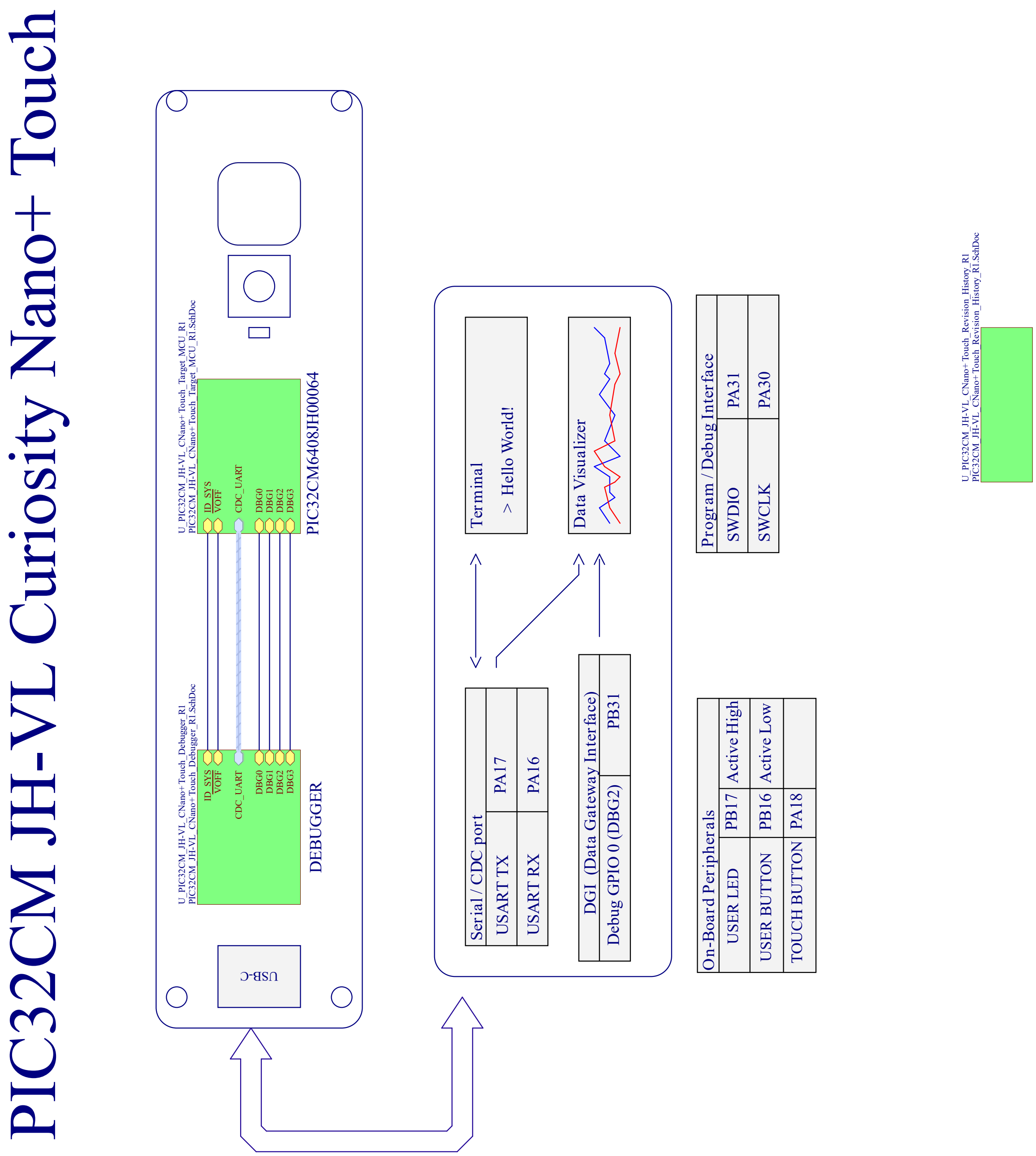


Figure 5-2. Power Supply and Debugger Schematic

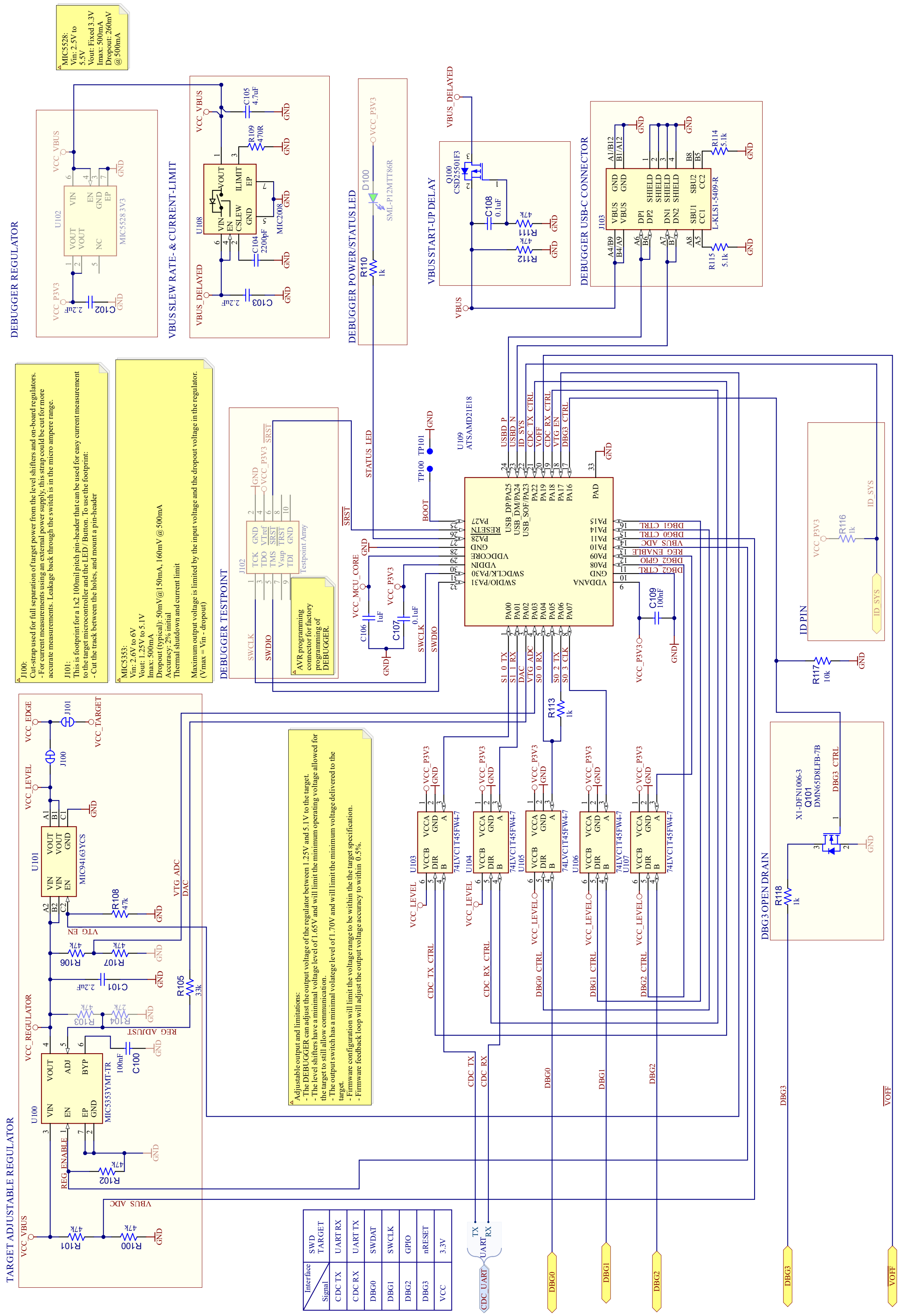
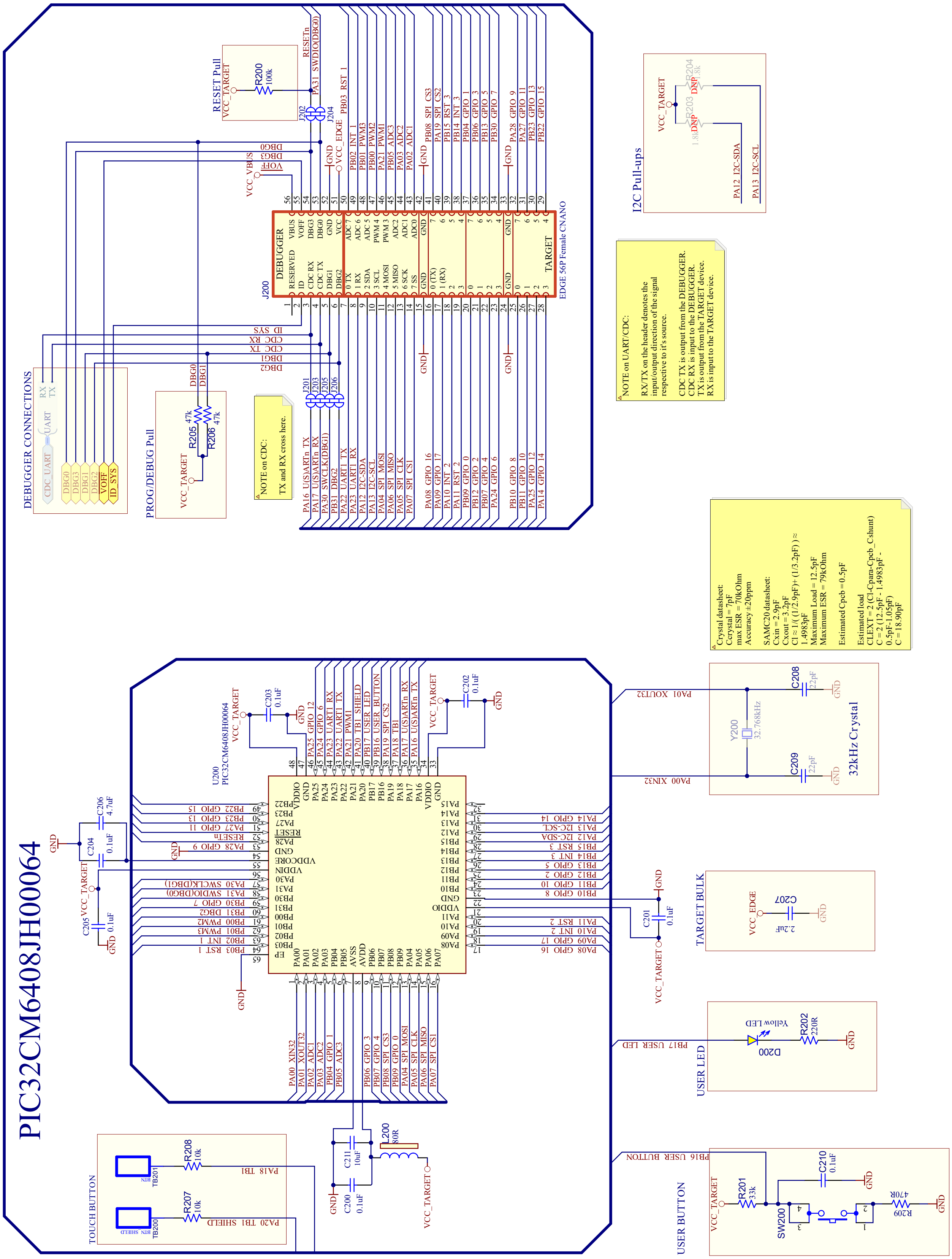


Figure 5-3. Target Microcontroller Schematic



5.2. Assembly Drawing

This section contains the following assembly drawings for the PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit:

- Top Assembly Drawing, Figure 5-4
- Bottom Assembly Drawing, Figure 5-5

Figure 5-4. Top Assembly Drawing

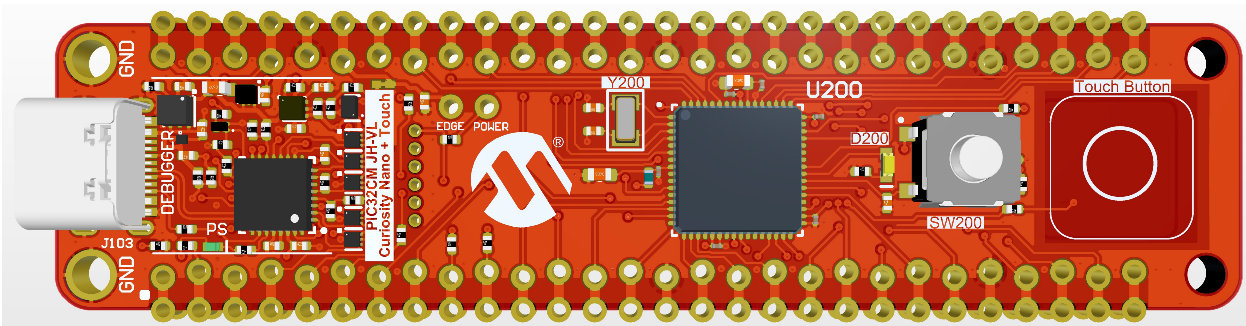
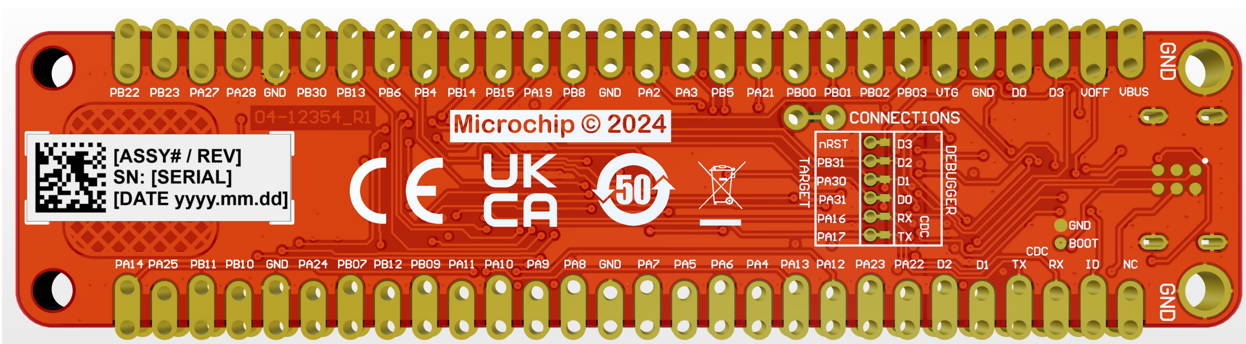


Figure 5-5. Bottom Assembly Drawing



5.3. Bill of Materials

Table 5-1. PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit Bill of Materials

Quantity	Designator	Description	Manufacturer 1	Manufacturer Part Number 1
5	C100, C107, C108, C109, C210	CAP CER 0.1 uF 16V 10% X7R SMD 0402	KEMET	C0402C104K4RACAUTO
4	C101, C102, C103, C207	CAP CER 2.2 uF 16V 10% X5R SMD 0402	Murata	GRM155R61C225KE44D
1	C104	CAP CER 2200 pF 25V 5% C0G SMD 0402	KEMET	C0402C222J3GACTU
2	C105, C206	CAP CER 4.7uF 10V 10% X5R SMD 0603	KEMET	C0603C475K8PACTU
1	C106	CAP CER 1uF 6.3V 10% X5R SMD 0402	KEMET	C0402C105K9PACTU
6	C200, C201, C202, C203, C204, C205	CAP CER 0.1uF 16V 10% X7S 0201	D3_Murata Electronics	GRM033C71C104KE14J
2	C208, C209	CAP CER 22pF 50V 5% NP0 SMD 0402	Murata	GRM1555C1H220JZ01D
1	C211	CAP CER 10 uF 10V 10% X5R SMD 0603	Samsung Electro-Mechanics	CL10A106KP8NNNC
1	D100	DIO LED YELLOW-GREEN 2.2V 20 mA 25 mcd Clear SMD 0402	Rohm Semiconductor	SML-P12MTT86R

Table 5-1. PIC32CM JH-Value Line Curiosity Nano+ Touch Evaluation Kit Bill of Materials (continued)

Quantity	Designator	Description	Manufacturer 1	Manufacturer Part Number 1
1	D200	LED YELLOW DIFFUSED 1608 SMD	Rohm Semiconductor	SML-D12Y1WT86
1	J103	CON USB2.0 Type-C Female SMD R/A	KLS Electronic	L-KLS1-5409-R
1	L200	FERRITE 80R@100 MHz 1.5A SMD 0402	Murata	BLM15PD800SN1D
1	LABEL1	LABEL PCBA 18x6 mm Datamatrix Assy# / Rev / Serial / Date	ACT Logimark AS	505462
1	Q100	TRANS FET P-CH CSD25501F3 -20V -3.6A 0.076R LGA	Texas Instruments	CSD25501F3
1	Q101	TRANS FET N-CH DMN65D8LFB-7B 260 mA, 60V, 430 mW X1_DFN1006-3	Diodes Incorporated	DMN65D8LFB-7B
11	R100, R101, R102, R103, R106, R107, R108, R111, R112, R205, R206	RES TKF 47k 1% 1/16W SMD 0402	YAGEO	RC0402FR-0747KL
1	R104	RES TKF 27k 1% 1/16W SMD 0402	Rohm Semiconductor	TRR01MZPF2702
2	R105, R201	RES TKF 33k 1% 1/16W SMD 0402	Yageo	AC0402FR-0733KL
2	R109, R209	RES TKF 470R 1% 1/16W MF 0402	Yageo	RC0402FR-07470RL
4	R110, R113, R116, R118	RES TKF 1k 1% 1/10W SMD 0402 AEC-Q200	Panasonic Electronic Components	ERJ-2RKF1001X
2	R114, R115	RES TKF 5.1k 1% 1/10W SMD 0402	Panasonic Electronic Components	ERJ-2RKF5101X
1	R117	RES TKF 10k 1% 1/10W SMD 0402	Panasonic	ERJ-2RKF1002X
1	R200	RES TKF 100k 1% 1/16W SMD 0402	Stackpole Electronics Inc	RMCF0402FT100K
1	R202	RES TKF 220R 1% 1/16W SMD 0402	Yageo	RC0402FR-07220RL
0	R203, R204	RES TKF 1.8k 1% 1/10W SMD 0402	Panasonic Electronic Components	ERJ-2RKF1801X
2	R207, R208	RES TF 10k 1% 1/10W SMD 0402 AEC-Q200	Vishay Beyschlag	MCS0402MC1002FE000
1	SW200	SWITCH TACT SPST 12 50 mA TS604VM1-035CR-R SMD	Dailywell Electronics Co.,Ltd.	TS604VM1-035CR-R
1	U100	MCHP ANALOG LDO ADJ 500 mA MIC5353YMT-TR MLF-6	Microchip Technology	MIC5353YMT-TR
1	U101	IC LOAD SWITCH HI SIDE 3A 6WLCSP	Microchip Technology	MIC94163YCS-TR
1	U102	MCHP ANALOG LDO 3.3V MIC5528-3.3YMT-TR 6-TDFN	Microchip Technology	MIC5528-3.3YMT-TR
5	U103, U104, U105, U106, U107	IC VOLTAGE TRANSLATOR BI-DIR 1 CIRCUIT 74LVC1T45FW4-7 X2-DFN1010-6	Diodes Incorporated	74LVC1T45FW4-7
1	U108	MCHP ANALOG POWER SWITCH 5.5V 2.1A MIC2008YML-TR VDFN-6	Microchip Technology	MIC2008YML-TR
1	U109	MCHP MCU 32-BIT 48 MHz, 256 Kb, 32 Kb ATSAMD21E18A-MUT QFN-32	Microchip Technology	ATSAMD21E18A-MUT
1	U200	MCHP MCU 32-BIT 48 MHz, 64 kB, 8 kB PIC32CM6408JH00064-I/5LX VQFN-64	Microchip Technology	PIC32CM6408JH00064-I/5LX
1	Y200	MCHP CRYSTAL 32.768 kHz, 9 pF SMD L3.2W1.5XH0.9	Microchip Technology	VMK3-9001-32K7680000TR

6. Revision History

Revision B - May 2025

The following updates was performed for this revision:

Preface	Updated figure.
Introduction	Updated Figure 1-1 in Board Overview section.
Getting Started	Updated Figure 2-2 in Using Pin-Headers section.
On-Board Debugger	Updated Figure 3-3 in Disconnecting the On-Board Debugger section.
Hardware	Update LED section.

Revision A - April 2025

This is the initial released version of this document.

Microchip Information

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

ISBN: 979-8-3371-1316-6

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Product Page Links

PIC32CM6408JH00064