MCCI Corporation
3520 Krums Corners Road
Ithaca, New York 14850 USA
Phone +1-607-277-1029
Fax +1-607-277-6844
www.mcci.com

Model 3421 Users Guide

Engineering Report 950001641
Rev C
Date: 2025-11-05

Copyright © 2025

All Rights Reserved

PROPRIETARY NOTICE AND DISCLAIMER

Unless noted otherwise, this document and the information herein disclosed are proprietary to MCCI Corporation, 3520 Krums Corners Road, Ithaca, New York 14850 ("MCCI"). Any person or entity to whom this document is furnished or having possession thereof, by acceptance, assumes custody thereof and agrees that the document is given in confidence and will not be copied or reproduced in whole or in part, nor used or revealed to any person in any manner except to meet the purposes for which it was delivered. Additional rights and obligations regarding this document and its contents may be defined by a separate written agreement with MCCI, and if so, such separate written agreement shall be controlling.

The information in this document is subject to change without notice, and should not be construed as a commitment by MCCI. Although MCCI will make every effort to inform users of substantive errors, MCCI disclaims all liability for any loss or damage resulting from the use of this manual or any software described herein, including without limitation contingent, special, or incidental liability.

MCCI, TrueCard, TrueTask, MCCI Catena, and MCCI USB DataPump are registered trademarks of MCCI Corporation.

MCCI Instant RS-232, MCCI Wombat, NerveCircuit, and InstallRight Pro are trademarks of MCCI Corporation.

All other trademarks and registered trademarks are owned by the respective holders of the trademarks or registered trademarks.

Copyright © 2025 by MCCI Corporation.

Document Release History

Rev A	2023-12-21	Initial Release
Rev B	2024-09-02	Added support for firmware update
Rev C	2025-11-05	Added support for control port command reference

TABLE OF CONTENTS

1 Preface 5
1.1 Relevant Documents 5
1.2 Glossary 5
2 Model 3421 Overview 6
3 Unpacking and Setup 6
4 Model 3421 BDP Application 7
4.1 BDP Firmware 7
4.1.1 Descriptor 7
4.2 Model 3421 Test Setup 8
4.2.1 Pre-requisites 8
Hardware: 8
Software: 8
4.2.2 Model 3421 Display Test Setup: 8
4.3 Model 3421 Test Procedure 9
5 Firmware Update 10
6 Control Port Command Reference 12
6.1 Main Commands and their usage 13
6.1.1 Get Device Mode 13
6.1.2 Set Device Mode 13
6.1.3 Controlling the SSC bit 13
6.1.4 Displaying the App version 14
6.2 Model3421 Shell Command Line Interface List 14
7 Model 3421 App Release Versions 18
7.1 v1.2.0 18
7.2 v1.0.0 18
8 Getting Help 18

LIST OF TABLES

Table 1. References, IP Blocks 5
Table 2. Model 3421 Shell Commands 14

LIST OF FIGURES

Figure 1. Model 3421 as USB Type-C to HDMI converter 6
Figure 2. Model 3421 BDP Demo Setup 8
Figure 3. Before BDP driver installation 9
Figure 4. After BDP driver installation 9
Figure 5 Model 3421 enumerated as Bootloader Device 10

LIST OF SEQUENCE DIAGRAMS

No table of figures entries found.

1 Preface

The MCCI® Model 3421 is a USB-to-HDMI converter device that is developed based on the USB/VESA joint specification.

1.1 Relevant Documents

The following documents are referenced by this document.

Table 1. References, IP Blocks

Reference	Description	File/ URL
[DataPump]	MCCI USB DataPump portable USB device firmware library	Web page https://mcci.io/datapump, technical overview https://mcci.io/971001018
[FTDI-Driver]	FTDIchip driver download page for control port of Model 3421	https://ftdichip.com/drivers/vcp-drivers/
[MA2085]	Intel Movidius Myriad X MA2085 VPU	Intel® Movidius™ Myriad™ X VPU Product Brief
[Minicom]	Terminal emulator program for Unix-like operating systems	https://salsa.debian.org/minicom-team/minicom
[NIST-Binary-Units]	NIST reference page: “Prefixes for binary multiples”	https://physics.nist.gov/cuu/Units/binary.html
[TeraTerm]	Tera Term home page	https://ttssh2.osdn.jp/index.html.en

1.2 Glossary

BDP	Basic Display Protocol

2 Model 3421 Overview

The MCCI® Model 3421 is a complete single-board USB device that performs as a USB to HDMI converter.

Figure 1. Model 3421 as USB Type-C to HDMI converter



3 Unpacking and Setup

The Model 3421 kit includes the following components:

- The Model 3421 board,
- A One-meter USB-C to USB-C cable,
- A three-foot Micro-HDMI to HDMI cable, and
- A three-foot USB-A plug to USB micro-B plug cable, for use with the remote-control port.

If you purchase the Model 3421 in bulk pack or as part of a larger set of test equipment, the provided accessories may be different.

4 Model 3421 BDP Application

4.1 BDP Firmware

USB Basic Display Protocol Application is developed based on USB/VESA Display Port Specification to support USB to HDMI conversion.

4.1.1 Descriptor

The following are the Device, Configuration, and Interface descriptors of Model 3421 as a BDP device.

Device Descriptor:

```
bcdUSB                  : 0x320 (USB Version 3.20)
bDeviceClass            : 0x00 (defined by the interface descriptors)
bDeviceSubClass         : 0x00
bDeviceProtocol         : 0x00
bMaxPacketSize0         : 0x09 (9 bytes)
idVendor                : 0x040E (MCCI Corporation)
idProduct               : 0xF64C
bcdDevice               : 0x0100
iManufacturer           : 0x01 (String Descriptor 1)
iProduct                : 0x02 (String Descriptor 2)
iSerialNumber           : 0x03 (String Descriptor 3)
bNumConfigurations      : 0x01 (1 Configuration)
```

Configuration Descriptor:

```
wTotalLength            : 0x0045 (69 bytes)
bNumInterfaces          : 0x01 (1 Interface)
bConfigurationValue     : 0x01 (Configuration 1)
iConfiguration          : 0x00 (No String Descriptor)
bmAttributes            : 0xE0
 D7: Reserved, set 1    : 0x01
 D6: Self Powered       : 0x01 (yes)
 D5: Remote Wakeup      : 0x01 (yes)
 D4..0: Reserved, set 0 : 0x00
MaxPower                : 0x0D (26 mA)
```

Interface Descriptor:

```
bInterfaceNumber        : 0x00 (Interface 0)
bAlternateSetting       : 0x00
bNumEndpoints           : 0x03 (3 Endpoints)
bInterfaceClass         : 0x13 (unknown)
bInterfaceSubClass      : 0x00
bInterfaceProtocol      : 0x00
iInterface              : 0x04 (String Descriptor 4)
 Language 0x0409        : "Model 3421 BDP Function"
```

4.2 Model 3421 Test Setup

This section describes the test setup and the prerequisites required to test the MCCI's Model 3421 as a USB to HDMI converter.

4.2.1 Pre-requisites

Below is the list of items required,

Hardware:

1. Windows Host Test Machine with USB Type-C port
2. Model 3421 (available in the Model 3421 Kit)
3. Standard USB Type-C to Type-C cable (available in the Model 3421 Kit)
4. Standard HDMI to Micro-HDMI cable (available in the Model 3421 Kit)
5. External Monitor with HDMI support

Software:

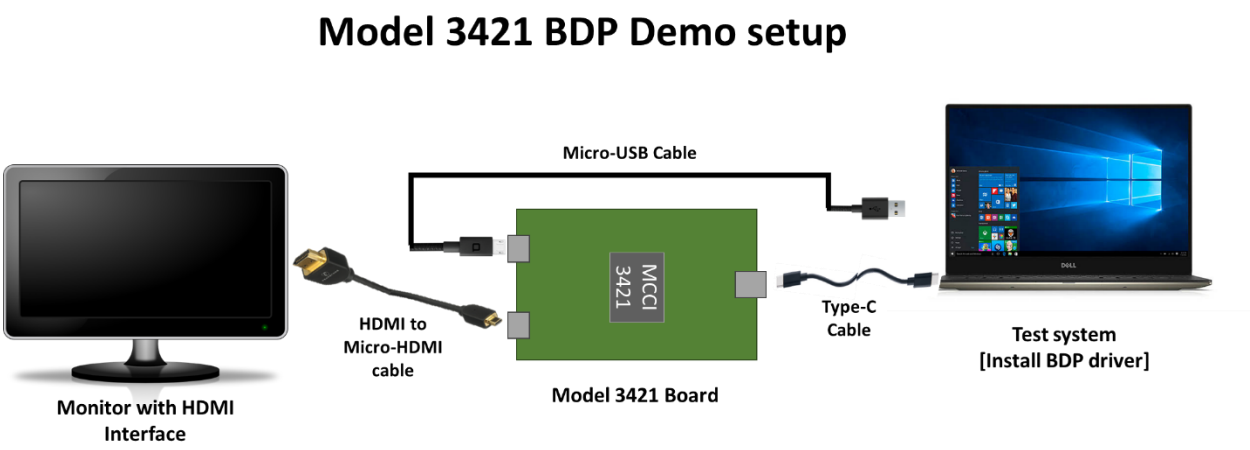
1. BDP driver

4.2.2 Model 3421 Display Test Setup:

The below figure explains to us the Model 3421 test setup to test the USB to HDMI converter.

The test objective is to extend the Host test machine's (Windows) Display to an External Monitor using the Model 3421.

Figure 2. Model 3421 BDP Demo Setup



The Model 3421 device's Type-C port must be connected to the Host Windows machine's Type-C port via the USB Type-C to Type-C cable.

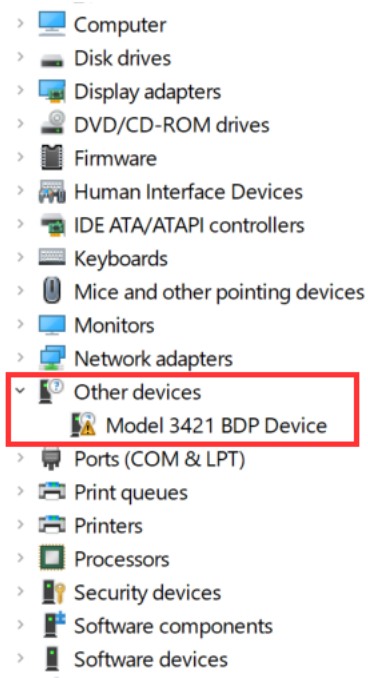
The Model 3421 device's Micro HDMI port must be connected to the External Monitor via the Micro-HDMI to HDMI cable.

4.3 Model 3421 Test Procedure

This section describes the step-by-step test procedure to test the Model 3421's BDP application to verify the USB to HDMI conversion.

1. Connect the Model 3421 device's Type-C port to the Host Windows machine's Type-C port via the USB Type-C to Type-C cable.
2. Allow the device to enumerate.
3. Check in the Device Manager under **`Other devices`** for **`Model 3421 BDP Device`** with a warning symbol as shown in the figure below.

Figure 3. Before BDP driver installation



4. Now, Install the BDP driver by right-clicking on the **`Model 3421 BDP Device`** and **`update driver`** option.
   NOTE: BDP driver can be availed from Microsoft/Synaptics, please contact MCCI to get one.
5. Once the BDP driver is installed successfully, the Model 3421 device comes up with the name **`BdpIddDriver Device`** under **`Display adapters`**.

Figure 4. After BDP driver installation



6. Now connect the Model 3421 device's Micro HDMI port to the External Monitor via the Micro-HDMI to HDMI cable.
7. Once the HDMI connection is made to the External Monitor, the Windows Machine's Display will be extended to the External monitor via Model 3421's USB to HDMI conversion.

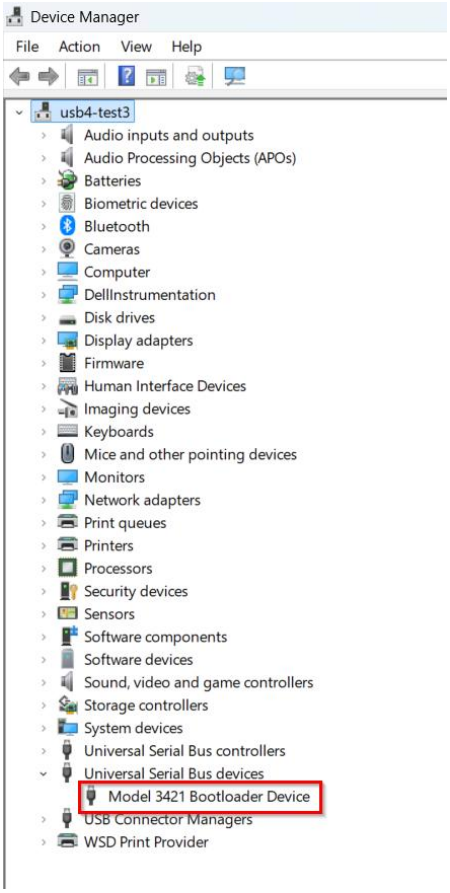
5 Firmware Update

One of the easiest ways to update the Firmware for Model 3421 is via DFU method.

To flash/update the MCCI BDP Application in the Model3421 device via DFU, follow the below steps,

1. Get a Windows machine.
2. The BDP Application DFU Firmware for Model 3421 can be downloaded from the MCCI Model 3421 Article Page, please search for the file `model3421-v1.2.0-dfu-update-package.zip` at the bottom of the article page. Download, extract and store it in a desired location.

Figure 5 Model 3421 enumerated as Bootloader Device



3. Open a command prompt and move to the location the DFU firmware is stored.

4. Copy the below DFU Firmware upgrade command in the prompt before connecting the Model 3421 device.
   `.\model3421-dfu-update.bat`
5. Because the device enumerates as `MCCI 3421 Bootloader Device` and stays in DFU mode only for approx. 8 seconds.
6. Connect the Model3421 to the Windows machine via the USB Type-C cable.
7. Now, run the below command to start the DFU firmware upgrade process (refer #4).
8. You will see like the below log,

```
dfu-util 0.11


Copyright 2005-2009 Weston Schmidt, Harald Welte and OpenMoko Inc.

Copyright 2010-2021 Tormod Volden and Stefan Schmidt

This program is Free Software and has ABSOLUTELY NO WARRANTY

Please report bugs to http://sourceforge.net/p/dfu-util/tickets/


Waiting for device, exit with ctrl-C

Opening DFU capable USB device...

Device ID 040e:0209

Device DFU version 0110

Claiming USB DFU Interface...

Setting Alternate Interface #0 ...

Determining device status...

DFU state(2) = dfuIDLE, status(0) = No error condition is present

DFU mode device DFU version 0110

Device returned transfer size 4096

Copying data from PC to DFU device

Download        [=========================] 100%      4089600 bytes

Download done.

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present
```

```
DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(7) = dfuMANIFEST, status(0) = No error condition is present

DFU state(2) = dfuIDLE, status(0) = No error condition is present

Done!
```

9. Once the DFU image is upgraded successfully, reboot the Model3421 device by disconnect and Connect via Type C port. Now, connect the Model3421 via Control port and verify the firmware version using the shell command `appversion` from the Terminal.

6 Control Port Command Reference

The control port command helps the user to understand the device/firmware and its properties, and set it to work according to their requirements.

To connect to the control port, you'll also need a terminal emulation program.

- On Windows, MCCI recommends Tera Term [TeraTerm].
- On macOS, MCCI suggests the `screen` program. Use a web search to get information; the basic command is:

```
screen /dev/tty.usbserial-FTxyz 115200
```

  Change *xyz* to the actual serial port assigned by macOS.

- On Linux, MCCI suggests the `minicom` program [Minicom]. Remember that on many distributions, you must add udev rules to set permissions on the created USB serial port, or else run `minicom` with root privileges.

Regardless of the terminal emulator you use, you should set up the terminal properties correctly.

- 115200 baud

- 8-bit data
- No parity
- One stop bit
- No local echo

Once this is set, you should be able to connect and get a command prompt, for example:

```
RTEMS Shell on /dev/console. Use 'help' to list commands.

SHLL [/] # help
help: The topics are
  all, model3421, help, misc, files, mem, rtems, monitor

SHLL [/] #
```

Each help topic has a set of commands defined in it. To get the commands associated with each topic, run `help <topic>`.

`help all` command should get the user all the available commands.

6.1 Main Commands and their usage

Below are some of the main Commands, their description, and usage given for our reference.

6.1.1 Get Device Mode

`getdevicemode` command helps to fetch the Model 3421 device's current application mode (loopback/bdp).

```
SHLL [/] # getdevicemode
Current device mode is bdp

SHLL [/] #
```

6.1.2 Set Device Mode

`setdevicemode` command helps to change the Model 3421 device's current application mode (loopback/bdp).

Note: The current version of firmware doesn't support **setdevicemode**, we are planning to add it in the upcoming release.

6.1.3 Controlling the SSC bit

`controlssc` command helps in setting the Spread Spectrum Clocking (SSC) bit in the corresponding USB PHY register to enable or disable it.

```
SHLL [/] # controlssc 0
-UsbPumpDataPumpRtemsApi_DeviceRestart:
UsbPhy SSC disabled

SHLL [/] # controlssc 1
-UsbPumpDataPumpRtemsApi_DeviceRestart:
UsbPhy SSC enabled
```

6.1.4 Displaying the App version

`appversion` displays the current available application version of Model 3421.

```
SHLL [/] # appversion


model3421-application v1.2.0
model3421-bdp-device v1.2.0
```

6.2 Model3421 Shell Command Line Interface List

Below is the list of Model 3421 shell command line interfaces and their descriptions.

Table 2. Model 3421 Shell Commands

Command	Description
setdebugflag	Command to set/clear the Debug Flags to see the resp. debug logs of Model3421 operations. `setdebugflag <object_name_pattern> <clear_debug_flag> <set_debug_flag>`
setdevicemode	Command to set the device application mode of Model3421, `setdevicemode` [0=bdp|1=loopback]
getdevicemode	Displays the current device application mode of Model 3421 (bdp or loopback)
controlssc	Enable/Disable the Spread Spectrum Clocking (SSC) bit in the USB PHY register. `controlssc` [0=disable|1=enable]
appversion	Displays the current available Model3421 application versions.
setgpio	setgpio [b|y|g] [1|0]
help	help [topic] # list of usage of commands
alias	alias old new
time	time command [arguments...]
cmdls	cmdls COMMAND...
cmdchown	cmdchown [OWNER][:[GROUP]] COMMAND...
cmdchmod	cmdchmod OCTAL-MODE COMMAND...
joel	joel [args] SCRIPT
date	date [YYYY-MM-DD HH:MM:SS]

echo	echo [args]
edit	edit [file ...]
sleep	sleep seconds [nanoseconds]
id	show uid, gid, euid, and egid
tty	show ttyname
whoami	show current user
logoff	logoff from the system
setenv	setenv [var] [string]
getenv	getenv [var]
unsetenv	unsetenv [var]
mdump	mdump [address [length [size]]]
wdump	wdump [address [length]]
ldump	ldump [address [length]]
medit	medit address value1 [value2 ...]
mfill	mfill address size value
mmove	mmove dst src length
cp	cp [-R [-H | -L | -P]] [-f | -i] [-pv] src target
pwd	pwd # print work directory
ls	ls [dir] # list files in the directory
chdir	chdir [dir] # change the current directory
mkdir	mkdir dir # make a directory
rmdir	rmdir dir # remove directory
chroot	chroot [dir] # change the root directory
chmod	chmod 0777 n1 n2... # change filemode
cat	cat n1 [n2 [n3...]] # show the ascii contents
mkrfs	mkrfs [-v] [-s blksz] [-b grpblk] [-i grpinode] [-I] [-o %inode] dev
mkdos	mkdos [-V label] [-s sectors/cluster] [-r size] [-v] path # format disk
msdosfmt	is an <alias> for command 'mkdos'
mv	[-fiv] source target ...
rm	[-f | -i] [-dIPRrvW] file ...
ln	ln ln [-fhinsv] source_file [target_file]
mknod	mknod mknod [-rR] [-F fmt] [-m mode] name [c | b] minor
umask	umask [new_umask]
lsof	lsof
mount	mount [-t type] [-r] [-L] [-o options] source target
unmount	unmount path # unmount disk
blksync	blksync driver # sync the block driver
blkstats	blkstats [-r|--reset] PATH_TO_DEVICE
fdisk	disk format and utility functions fdisk DISK_NAME prints the partition table fdisk DISK_NAME [FS N1 [N2 ...]] ... [write] [FORMAT]

	creates a new partition table fdisk DISK_NAME register creates a logical disk for each partition of the disk fdisk DISK_NAME unregister deletes the logical disks associated with the partitions of the di sk fdisk DISK_NAME mount mounts the file system of each partition of the disk fdisk DISK_NAME unmount unmounts the file system of each partition of the disk option values: DISK_NAME: absolute path to disk device like '/dev/hda' N*: weights of positive integers FS: 0x00 ... 0xff, fat12, fat16, fat32, data write: write the new partition table to the disk FORMAT: mbr [[no]dos], gpt
dd	dd [OPERAND]...
hexdump	hexdump [-bcCdovx] [-e fmt] [-f fmt_file] [-n length] [-s skip] [file ...]
debugrfs	debugrfs [-h]
df	[-hB] h human-readable output B scale sizes by SIZE before printing them
md5	md5 [file ...]
shutdown	shutdown
cpuinfo	cpuinfo
cpuuse	[-r] print or reset per thread cpu usage
top	[-r] print or reset per thread cpu usage
stackuse	print per thread stack usage
perioduse	[-r] print or reset per period usage
profreport	profreport
wkspace	Report on RTEMS Executive Workspace
malloc	malloc [walk]
rtrace	rtrace [-l]
rtc	real time clock read and set rtc prints the current time of day rtc YYYY-MM-DD [HH:MM:SS [TICKS]] sets the time of day and real time clock

spi	simple SPI read/write spi [-loh] [-c <cs>] [-s <speed>] [-m <mode>] <SPI_BUS> xx [xx [..]] <SPI_BUS> Bus device to use xx Hex value of a byte to send -c <cs> Use chip select <cs> (default: None) -m <mode> Use SPI mode <mode> (default: 0) -l Send LSB first -o Use loopback mode -s <speed> Bus speed in hz -h Print this help
i2cdetect	i2cdetect <I2C_BUS> try to detect i2c devices on the given bus
i2cget	i2cget <I2C_BUS> <CHIP-ADDRESS> <DATA-ADDRESS> [<NR-BYTES>] Get one or more bytes from an EEPROM like i2c device. Note that multiple bytes will be read in continuous mode.
i2cset	i2cset <I2C_BUS> <CHIP-ADDRESS> <DATA-ADDRESS> <VALUE> [<VALUE> [...]] set one byte of an EEPROM like i2c device
dir	is an <alias> for command 'ls'
cd	is an <alias> for command 'chdir'
exit	is an <alias> for command 'logoff'
config	Show the system configuration.
itask	List init tasks for the system
mpci	Show the MPCI system configuration, if configured.
pause	Monitor goes to `sleep` for specified ticks (default is 1). Monitor will resume at end of period or if explicitly awakened pause [ticks]
continue	Put the monitor to sleep waiting for an explicit wakeup from the program running.
go	Alias for 'continue'
symbol	[symbolname [symbolname ...]] Display value associated with specified symbol. Defaults to displaying all known symbols. symbol
extension	extension [id [id ...]] Display information about specified extensions. Default is to display information about all extensions on this node.
task	task [id [id ...]] Display information about the specified tasks. Default is to display information about all tasks on this node.
queue	Display information about the specified message queues. Default is to display information about all queues on this node. queue [id [id ...]]
sema	sema [id [id ...]] display information about the specified semaphores Default is to display information about all semaphores on this node
region	region [id [id ...]]

	display information about the specified regions Default is to display information about all regions on this node
part	part [id [id ...]] display information about the specified partitions Default is to display information about all partitions on this node
object	object [id [id ...]] Display information about specified RTEMS objects. Object id's must include 'type' information. (which may normally be defaulted)
driver	driver [major [major ...]] Display the RTEMS device driver table.
dname	Displays information about named drivers.
fatal	fatal [status] 'exit' with fatal error; default error is RTEMS_TASK_EXITTED
reset	(SW)Resets the System.
pthread	pthread [id [id ...]] Display information about the specified pthreads. Default is to display information about all pthreads on this node.

7 Model 3421 App Release Versions

This section lists the changes made in each release version of the Model 3421 BDP Application.

7.1 v1.2.0

Added CMX DMA support

7.2 v1.0.0

Initial Release with USB-to-HDMI support.

8 Getting Help

If you have a question about using the Model 3421, please visit MCCI's support community for the Model 3421. Feel free to post a question! We'll do our best to assist, and you may benefit from the experience of others. You may also post private questions to MCCI by opening a ticket or by sending an email to techsupport@mcci.com. Please be sure to include the serial number of your unit, which is found on the underside of the larger board. It is a hexadecimal string of the form 00-02-CC-xx-yy-zz.