KSZ8441HL Evaluation Kit User Guide

Revision 0.2 / August 15, 2013

Table of Contents

List of Figures

Introduction

The KSZ8441HL Evaluation Kit provides network designers the opportunity to explore the functionality and features of Micrel's KSZ8441 IEEE 1588 Precision Time Protocol (PTP) Enabled End Point IC. The KSZ8441HL is a highly integrated 1-port 10/100Base-T/TX/FX managed Ethernet end point with a second port for interfacing to a host processor. It is an ideal solution in industrial applications where clock synchronization across an Ethernet network is needed.

This KSZ8441HL Evaluation Kit User Guide provides the information necessary to set up and use the KSZ8441HL evaluation platform.

1 Tour of the Kit Contents

The KSZ8441 Evaluation Kit consists of the following components:

Hardware

- KSZ8441HLI-EVAL
    - Evaluation board with KSZ8441HLI 10/100 Ethernet end point
    - Board silkscreen: "KSZ8462HLL DEMO BOARD"
- KSZ9692-MII-PTP-EV[1]
    - Evaluation board with KSZ9692PB SOC, pre-loaded with Linux and PTP software
    - Board silkscreen: "Soc 2-MII BOARD"

Software – pre-loaded on the KSZ9692 board

- IEEE 1588 PTP stack – one or more
- IEEE 802.1AS (AVB) stack
- ***ptp_cli***
    - A utility for KSZ8441HL control and register access; command line interface
- ***ptp***
    - A utility for generating PTP test packets

Hardware Documentation

• KSZ8441HL Evaluation Kit User Guide	This document; it is the top level kit document
• KSZ8441HL Evaluation Board User Guide	Documents the KSZ8441HLI-EVAL board
• KSZ8462HL evaluation board schematic	
• KSZ8462HL evaluation board bill of materials (BOM)	
• KSZ8441HL IBIS model	
• KSZ9692 (SOC) board schematic	

Software Documentation

- Micrel KSZ8462HLI/KSZ8441HLI Step-by-Step Programmer's Guide
- Micrel 1588 PTP Application Notes
- Micrel 1588 PTP Developer Guide
- Micrel AVB/PTP Setup Guide
- Micrel PTP Utilities User Guide
- Micrel STP Usage Guide
- "README" file included with KSZ9692-MII-PTP-EV pre-loaded PTP software

The KSZ8441HLI-EVAL and KSZ9692-MII-PTP-EV are intended to be used together as shown in Figure 1, but must be ordered separately. The KSZ9692 board – also known as the "SOC board" – is not required for evaluation of the KSZ8441, but most customers use the two boards together. As an alternative to the KSZ9692 board, the user will need to interface the KSZ8441 board to another processor, and will need to port or develop the necessary software on that processor.

Notes:

1. The predecessor to the KSZ9692-MII-PTP-EV was the KSZ9692PB-PTP-EVAL. For most purposes, including all descriptions in this document, they are identical. The KSZ9692-MII-PTP-EV is silkscreened with “Soc 2-MII BOARD”. The KSZ9692PB-PTP-EVAL board has the silkscreen name “Soc Test BOARD”.

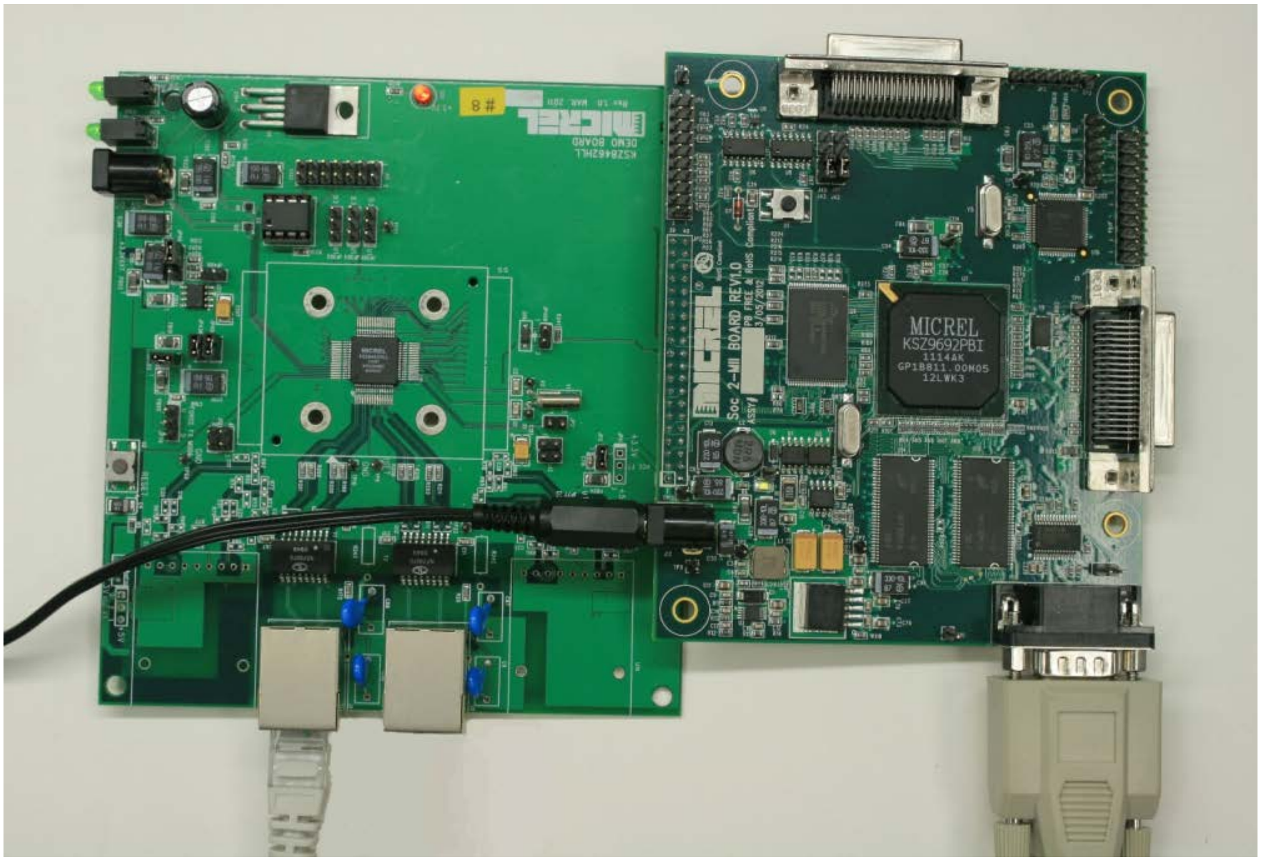


Figure 1 KSZ8441HLI-EVAL Eval Board and KSZ9692-MII-PTP-EV SOC Board

2 Setting up the KSZ8441HL Evaluation Platform

During set up of the evaluation platform, the reader may wish to have available the KSZ8441HL Eval Board User Guide and the KSZ8441HL evaluation board schematic. This will make it easier to locate key components requiring probing, configuration and/or modification.

2.1 Introduction

The IEEE1588 Precision Time Protocol (PTP) standard defines a network protocol that allows a number of slave network nodes to time synchronize to a master timing reference. The KSZ8441 has one 10/100 Ethernet network port plus a second interface for connection to a management host which can be a PTP slave (or master) node. The IEEE1588 standard refers to these masters and slaves as “ordinary clocks”.

The PTP protocol is implemented as a combination of hardware and software. The KSZ8441 hardware features include a PTP clock and packet timestamping capabilities in both ingress and egress directions. It also has a general purpose I/O (GPIO) block that is linked to the internal PTP clock for timestamping of input signals, and for generation of PTP timed output signals.

Any implementation of PTP also requires a PTP software stack running on a host processor. For evaluation purposes, Micrel provides Linux, multiple PTP stacks, and an AVB (IEEE802.1AS) stack pre-installed on the KSZ9692 (SOC) board. The KSZ9692 is a system on chip (SOC) with an ARM core. The KSZ9692-MII-PTP-EV and KSZ8441HLI-EVAL boards fit directly together to create one node in a PTP network. The network may consist of additional KSZ8441 / KSZ9692 nodes and/or other PTP nodes such as a dedicated IEEE 1588 grandmaster clock source, or an IEEE 1588 switch.

A PTP network must have a grandmaster clock source. (For KSZ8441 evaluation, the PTP terms *master* and *grandmaster* can be used interchangeably.) If more than one node advertises itself as a master, the PTP network has an algorithm (the best master clock algorithm) that will dynamically select one node as the master, with all other nodes reverting to being slaves. This algorithm compares several variables in a hierarchical manner. The two highest priority variables are *priority1* and *clockClass*. The last deciding variable, when all other variables are indecisive, is the MAC address. In this case, the device with the lowest MAC address will be selected as the master. By default, the SOC board has *priority1* = 128, and *clockClass* = 248, which is relatively low priority. GPS-based 1588 masters typically have stronger (numerically lower) values of *priority1* and *clockClass*, which will prevent a KSZ8441 / SOC node from being selected as master when both types of devices are on the same network. It is also possible to block a KSZ8441 / SOC from being a master by setting a profile variable. This is discussed in section 4.5.2.

It is also important to note that each node in the PTP network, as in any Ethernet network, must have MAC and IP addresses that are unique within that network. SOC board network addresses are set in software, and all boards are shipped with the same software. Thus, the default MAC and IP addresses are the same for each SOC board. Changing the network addresses is discussed in sections 4.5.1 and 4.5.2.

2.2 Basic Hardware Setup Options

Several configurations are available for verifying the general functionality of the components. Four configurations are shown below.

Every setup requires a communication connection between a computer (PC) and each SOC board. The PC is used as a control terminal. Two communication methods are available. Once set up, both methods provide the same access to Linux running on the SOC board.

1. Each SOC board has a 9-pin serial port. (For PC’s without a serial port, a USB-to-serial port converter can be used for interfacing a USB port on the PC to the serial port on the SOC board.) This method is shown in setups #1 and #2, and is described in detail in section 4.2.

2. Alternatively, the SOC board(s) may be accessed through the Ethernet port on the KSZ8441 board(s). This method is advantageous in situations where the network has multiple KSZ8441 / SOC nodes, and hence communication to all SOC devices can take place over one link to one PC. However, it requires setting up compatible IP addresses on the SOC boards and on the PC.

To do so, it may still be useful to use the serial interface connection method at least once, since it is not dependent on IP or MAC addressing. The Ethernet connection method is shown in setups #3 and #4, and is also detailed in section 4.3.

Another key factor is the choice of PTP grandmaster. One option is to use a dedicated 1588 PTP grandmaster, which is typically GPS referenced. This option is utilized in setups #1 and #3. The other option is to use a KSZ8441 / SOC node as a master. This is not commonly done in real networks, but is a viable option for evaluation purposes. Setups #2 and #4 utilize a KSZ8441 / SOC master.

2.2.1 Setup #1

Setup #1 uses a GPS-based time server as the grandmaster clock for the PTP network. It is the source of the clock information that is distributed to the KSZ8441 / SOC board pair. A minimum configuration utilizes a single KSZ8441 / SOC pair as shown in Figure 2. To add additional KSZ8441 / SOC pairs, all network nodes will need to be connected to a separate IEEE 1588 enabled switch.

The PC connects to the SOC board via separate serial port connection. The RS-232 serial interface on the SOC board is described in section 4.2. The software that is pre-loaded on the SOC board is set up to configure the KSZ8441's GPIO6 pin as a "1 pulse per second" (1PPS) output signal. In all of these setups, an oscilloscope is used to view the jitter and offset between the 1PPS signals generated from the KSZ8441 slave node and the grandmaster.

If there is more than one KSZ8441 node in the network, the user must change the IP and MAC addresses in the configuration file of each SOC board so that they are unique within the network. For details see sections 4.5.1 and 4.5.2.

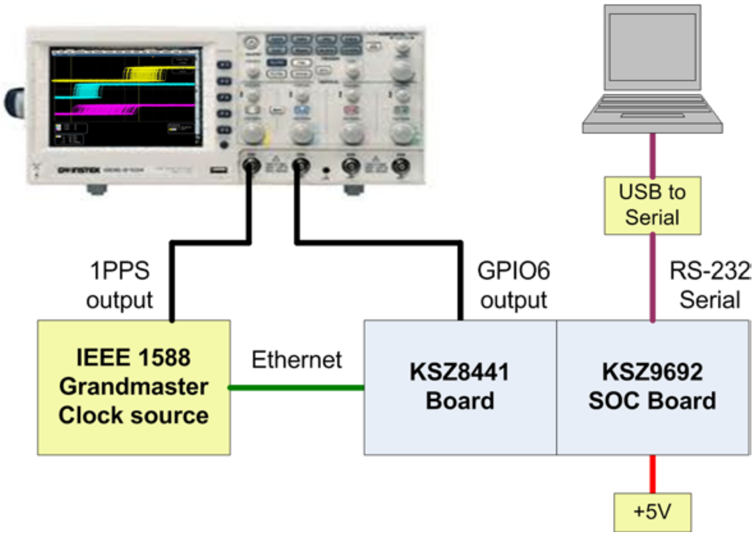


Figure 2 Setup #1

2.2.2 Setup #2

Instead of using a specialized time server as the grandmaster clock, setup #2 uses a second KSZ8441 / SOC pair as the grandmaster. The free-running onboard crystal on the master KSZ8441 eval board is the timing reference. There is no conventional real time clock on the KSZ8441 board, so the actual time reference that it provides is of no practical use. Even so, it offers a useful and convenient way to investigate PTP behavior without the use of a more expensive GPS based grandmaster clock source. All aspects of the PTP protocol are still utilized and valid, even though the time value being broadcast by the KSZ8441 master is not accurate.

With earlier versions of the KSZ9692 PTP software, the PTP configuration file needs to be reconfigured to enable at least one KSZ8441 node to be a PTP master. This involves clearing the DDS_SLAVE_ONLY variable, and is described in section 4.5.2. In current and future versions of the SOC software, this variable is already cleared by default, so no action is required to enable a node to be a master.

Each SOC board must be configured with locally unique IP and MAC addresses. In a network that consists only of KSZ8441 / SOC nodes, with no change to the PTP master priority variables, the SOC board with the lowest MAC address will become the PTP network master.

As with all of these setup options, more KSZ8441 / SOC board pairs can be added to the network only by connecting all devices into an IEEE 1588 enabled switch.

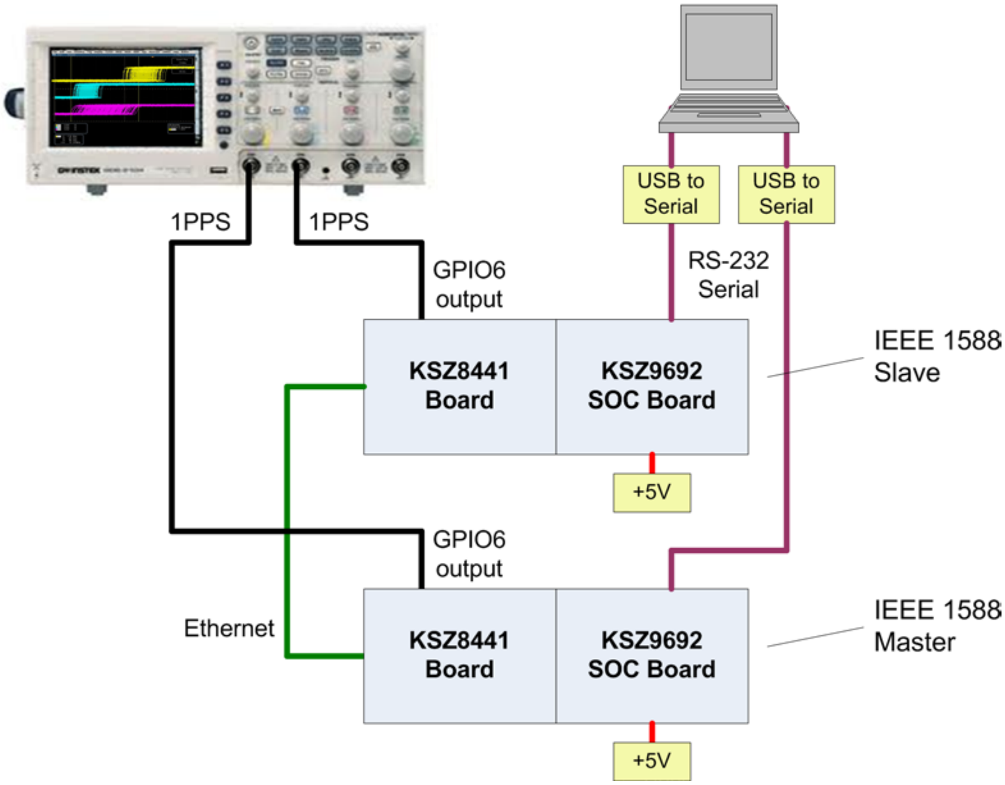


Figure 3 Setup #2

2.2.3 Setup #3

Like setup #1, setup #3 uses a GPS-based time server as the grandmaster clock for the PTP network. The difference in this setup is that the PC uses an Ethernet connection rather than an RS-232 serial cable to communicate to the KSZ8441 / SOC node. An IEEE 1588 enabled switch is needed for connecting all of these devices together. Telnet is used for PC-to-SOC communication, which requires that the PC and the SOC boards be configured with compatible IP addresses as described in section 4.3.

Figure 4 shows one KSZ8441 network node, but additional KSZ8441 nodes may be connected to the switch.

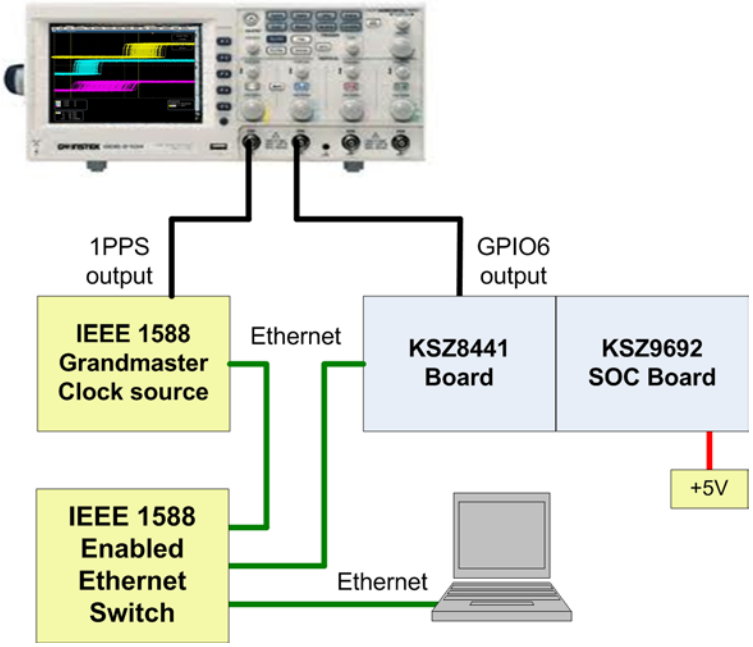


Figure 4 Setup #3

2.2.4 Setup #4

Setup #4 uses a second KSZ8441 / SOC pair as the grandmaster. The free-running onboard crystal on the master KSZ8441 eval board is the timing reference. There is no conventional real time clock on the KSZ8441 board, so the actual time reference that it provides is of no practical use. Even so, it offers a useful and convenient way to investigate PTP behavior without the use of a more expensive GPS based grandmaster clock source. All aspects of the PTP protocol are still utilized and valid, even though the time value being broadcast by the KSZ8441 master is not accurate.

With earlier versions of the KSZ9692 PTP software, the PTP configuration file needs to be reconfigured to enable at least one KSZ8441 node to be a PTP master. This involves clearing the DDS_SLAVE_ONLY variable as described in section 4.5.2. In current and future versions of the SOC software, this variable is already cleared by default, so no action is required to enable a node to be a master.

Each SOC board must be configured with locally unique IP and MAC addresses, as described in sections 4.5.1 and 4.5.2. In a network that consists only of KSZ8441 / SOC nodes, with no change to the PTP master priority variables, the SOC board with the lowest MAC address will become the PTP network master.

Setup #4 uses the Ethernet method of PC-to-SOC communication. Once unique network addresses have been configured, the PTP network is assembled by connecting all nodes into an IEEE 1588 enabled switch. Use Telnet as the communication protocol, as detailed in section 4.3.

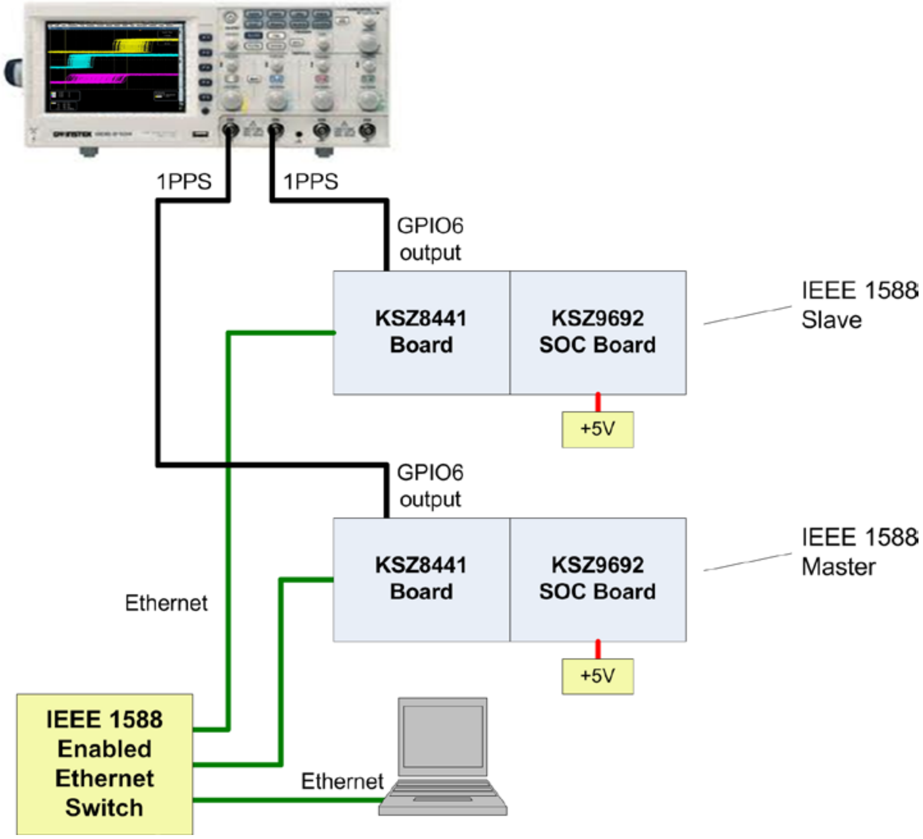


Figure 5 Setup #4

3 Quick Start Guides

Below are quick start guides for each of the four setups described in section 2.2. The intention is to guide the user from receipt of the Micrel evaluation board(s) to a functioning evaluation PTP network in the shortest possible time. Note that the quick start guides are only an outline of the necessary steps. Additional details are provided in sections 4 and 5.

3.1 Setup #1 Quick Start

1. Set up the IEEE1588 grandmaster time server for the following PTP variables:
   - End-to-end (E2E) delay mechanism
   - IPv4 multicast addressing
2. For each pair of KSZ8441 and SOC boards:
   a. Connect together the KSZ8441 and SOC boards and power them up. For details, see section 4.1.

   b. Use a null-modem serial cable to connect a PC to the DB-9 connector on the SOC board. Open a communication window and establish a connection with the SOC. See section 4.2 for details.

   c. If more than one KSZ8441 / SOC pair will be used, configure each one with locally unique IP and MAC addresses. See sections 4.5.1 and 4.5.2 for details. If only one KSZ8441 / SOC pair is used, this step may be skipped.

   Note: Step 2c is a one-time procedure for new boards or boards with updated software. Once new addresses have been assigned, this step does not need to be performed again.
3. Connect together the Ethernet ports of the IEEE1588 grandmaster clock source and the KSZ8441 board. Use an IEEE 1588 switch if there is more than one KSZ8441 node.
4. Connect the 1PPS outputs of the IEEE1588 grandmaster and each KSZ8441 board to the oscilloscope. See section 4.4 for details.
5. In the terminal window for each KSZ8441 / SOC node, type the following command to start the default PTP software stack: ***etc/ptpinit -d***
6. Observe the 1PPS signals on the oscilloscope.

3.2 Setup #2 Quick Start

1. For each pair of KSZ8441 and SOC boards:
   a. Connect together the KSZ8441 and SOC boards and power them up. For details, see section 4.1.

   b. Use a null-modem serial cable to connect a PC to the DB-9 connector on the SOC board. Open a communication window and establish a connection with the SOC. See section 4.2 for details.

   c. Configure each KSZ8441 / SOC board pair with locally unique IP and MAC addresses. See sections 4.5.1 and 4.5.2 for details.

   Note: Step 1c is a one-time procedure for new boards or boards with updated software.

Once new addresses have been assigned, this step does not need to be performed again.

2. Connect together the Ethernet ports of the two KSZ8441 boards. Use an IEEE 1588 switch if there are more than two KSZ8441 nodes.
3. Connect the 1PPS output of each KSZ8441 board to the oscilloscope. See section 4.4 for details.
4. In the terminal window for each KSZ8441 / SOC node, type the following command to start the default PTP software stack: ***etc/ptpinit -d***
5. Observe the 1PPS signals on the oscilloscope.

3.3 Setup #3 Quick Start

1. Set up the IEEE1588 grandmaster time server for the following PTP variables:
    - E2E delay mechanism
    - IPv4 multicast addressing
2. If more than one KSZ8441 / SOC pair will be used, configure each one with locally unique IP and MAC addresses. If only one KSZ8441 / SOC pair is used, this step may be skipped. Perform step 2 on one KSZ8441 / SOC pair at a time, <u>before</u> connecting multiple pairs together into a network.

   Note: Step 2 is a one-time procedure for new boards or boards with updated software. Once new addresses have been assigned, this step does not need to be performed again.

    a. Connect together the KSZ8441 and SOC boards and power them up. For details, see section 4.1.

    b. Use an Ethernet cable to connect a PC to the Ethernet port on the KSZ8441 board. Open a communication window and establish a Telnet session with the SOC board. See section 4.3 for details.

    c. Configure each SOC board with locally unique IP and MAC addresses. See sections 4.5.1 and 4.5.2 for details.
3. If not already done, assemble and power up the KSZ8441 and SOC boards.
4. Use an IEEE 1588 Ethernet switch to connect together the Ethernet ports of the IEEE1588 grandmaster clock source, the KSZ8441 board(s), and the PC.
5. Establish a Telnet session on the PC to each KSZ8441 / SOC node in the network.
6. Connect the 1PPS output of the IEEE1588 grandmaster and each KSZ8441 board to the oscilloscope. See section 4.4 for details.
7. In the terminal window for each KSZ8441 / SOC node, type the following command to start the default PTP software stack: ***etc/ptpinit -d***
8. Observe the 1PPS signals on the oscilloscope.

3.4 Setup #4 Quick Start

1. Configure each SOC board with locally unique IP and MAC addresses. Perform the following steps on one KSZ8441 / SOC pair at a time, <u>before</u> connecting multiple pairs together into a network.

Note: Step 1 is a one-time procedure for new boards or boards with updated software. Once new addresses have been assigned, this step does not need to be performed again.

a. Connect together the KSZ8441 and SOC boards and power them up. For details, see section 4.1.

b. Use an Ethernet cable to connect a PC to the Ethernet port on the KSZ8441 board. Open a communication window and establish a Telnet session with the SOC board. See section 4.3 for details.

c. Configure each SOC board with locally unique IP and MAC addresses. See sections 4.5.1 and 4.5.2 for details.

2. If not already done, assemble and power up the KSZ8441 and SOC boards.
3. Use an IEEE 1588 Ethernet switch to connect together the Ethernet ports of the IEEE1588 grandmaster clock source, the KSZ8441 board(s), and the PC as shown in Figure 5.
4. Establish a Telnet session on the PC to each KSZ8441 / SOC node in the network.
5. Connect the 1PPS output of each KSZ8441 board to the oscilloscope. See section 4.4 for details.
6. In the terminal window for each KSZ8441 / SOC node, type the following command to start the default PTP software stack: ***etc/ptpinit -d***
7. Observe the 1PPS signals on the oscilloscope.

4 Hardware and Software Details

This section describes the various features and details of the KSZ8441 evaluation kit hardware and software.

4.1 Connecting and powering the KSZ8441 and SOC boards

The SOC board connects to the host interface of the KSZ8441 board. This interface is used by the processor on the SOC board to communicate with the KSZ8441 device to perform in-band register read and write operations, as well as transferring Ethernet traffic to and from the processor.

The host interface is located on connector J16 on the KSZ8441 Eval Board and on JP2 on the SOC board. Simply mate these two 40-pin connectors as shown in Figure 1. Because the header on the KSZ8441 board is not shrouded or keyed, care should be taken to ensure correct orientation and alignment of the connectors. Additionally, the boards should be unpowered when the connection is made.

Both boards have a power connector for a 5V power supply, but it possible to power both boards through a single power connector. To do so, a jumper must be installed on header JP3 on the KSZ8441 board. Because the SOC board draws more power, it is preferred to apply power to the SOC board rather than the KSZ8441 board.

The power jack on the KSZ8441 and SOC boards has an inner diameter of 2.5mm. A 2.1mm plug will not fit. Note that 2.5mm plugs are compatible with both 2.5mm and 2.1mm jacks.

When a jumper is not installed on JP3, each board must be powered individually. Because the SOC software runs a routine at power up to initialize the KSZ8441, it is suggested that the KSZ8441 board be powered up either before the SOC board, or at the same time. Alternatively, the SOC board can be reset – either from a terminal command or by pressing the reset button – once the KSZ8441 board is powered up.

Power requirements

• KSZ8441 board	250mA
• SOC board	650mA

The KSZ8441 current requirement assumes that the board is configured with an UTP copper (RJ-45) interface. If the port is configured for fiber instead of copper, the current requirement for the fiber module must be added to the requirement above.

The boards can be manually reset by pressing the reset pushbutton switch located on each board. The proper order of resetting should be the KSZ8441 board first, followed by the SOC board.

4.2 RS-232 Serial Port Communication Method

Serial port communication between the user PC and the KSZ8441 / SOC boards is convenient and does not require knowledge of the IP address on the SOC board, but requires a separate cable for each KSZ8441 / SOC pair. The following section describes an alternative communication method.

1. Connect a null-modem cable from the PC's serial port (or a USB port) to the SOC board's serial port.

2. Run a terminal emulation program on the PC. Many programs are available, including PuTTY and TeraTerm.

3. Configure your PC's terminal emulation program to the following serial port settings:

Baud rate:	115,200
Data bits:	8
Parity:	none
Stop bits:	1

4. Once connected, press the **Enter** key on your keyboard to get a command prompt, or press the reset button on the SOC board.

5. From the Linux command prompt, "`/ #`", you can access configuration files, or run a utility or a protocol stack.

4.3 Ethernet Communication Method

Ethernet is an alternative method for communication between the SOC boards and a PC. It has an advantage over serial port communication because only a single Ethernet connection is required at the PC for simultaneous communication to all the networked KSZ8441 / SOC nodes. Connection setups are shown in Figure 4 and Figure 5.

Note: All new (or newly software updated) SOC boards have the same IP and MAC addresses. For such boards, the Ethernet communication method may be used to connect to one KSZ8441 / SOC board pair at a time. Once the IP and MAC addresses have been individualized on each SOC board, they may be networked together, and the PC can simultaneously communicate with all of them.

1. Set up the PC with an IPv4 address that is compatible with the SOC IP address. For the default settings shown below, the PC must have an IP address of `10.xxx.xxx.xxx` where "xxx" is any valid value. (Also check the IP mask of the PC!)

Default SOC network settings:

IP address:	`10.32.2.80`
IP address mask:	`255.0.0.0`

[For earlier revisions of the SOC software, the default IP address was `10.32.2.27`]

2. Connect an Ethernet cable between the PC and an unused port on a KSZ8441 board.

3. Open a command window on a Windows PC, or its equivalent. A Telnet session can be established by typing **`telnet 10.32.2.80`** (or the appropriate address). When the Telnet session is opened, enter the following login name and password when prompted:

Login:	**`root`**
Password:	**`micrel`**

4. At the first prompt, "**`~ #`**", type "**`cd /`**" or "**`cd ..`**" to get to the main Linux user file directory.

5. From the command prompt, "**`/ #`**", you can access configuration files, or run a utility or a protocol stack.

6. When the PC is connected to a network with more than one KSZ8441 / SOC node, a separate Telnet session (each in its own command window) is needed for each node.

Troubleshooting Tips:

- From the command window on the PC, use the **`ping`** command to test for communication to a specific IP address.
- From the command window on the PC, use the **`ipconfig`** command to find the network setting for the PC. It may only show information for ports that have live physical connections.
- If the IP address of the KSZ9692 is not known, a Telnet connection cannot be established. In this case, use the serial interface as described in the previous section.

4.4 Monitoring the 1PPS Output

At boot up, the software on the SOC board configures the KSZ8441 to output a 1PPS signal on GPIO6. The 1PPS signal is generated from the internal PTP clock within the KSZ8441. When a PTP software stack is running and the KSZ8441 / SOC pair are locked to a master, the 1PPS output should be closely synchronized to the 1PPS signal from the master. When not locked to a master, the PTP clock inside the KSZ8441 is running open loop from the local 25MHz reference clock. All of the GPIO pins are brought out on header J15. GPIO6 is located on J15, pin 13, as highlighted in Figure 6.

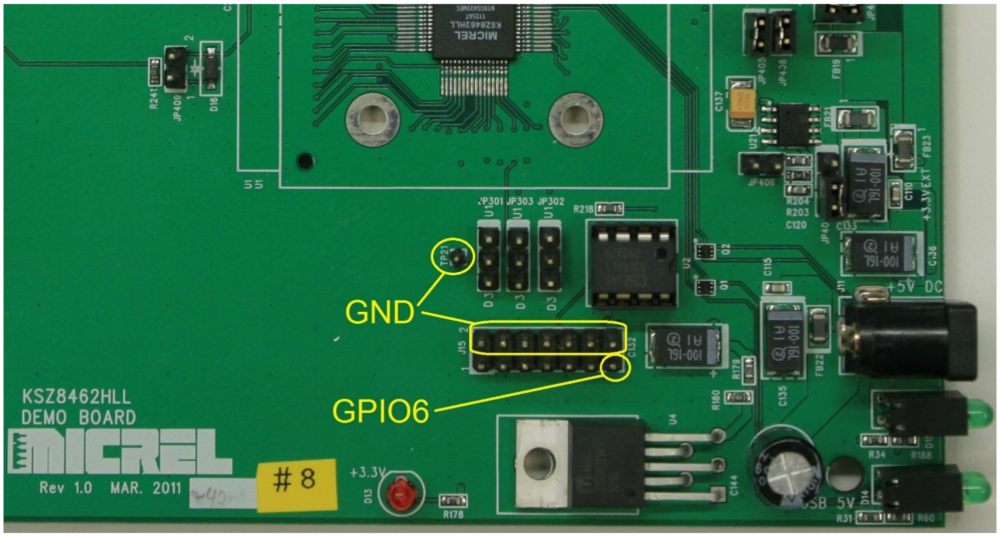


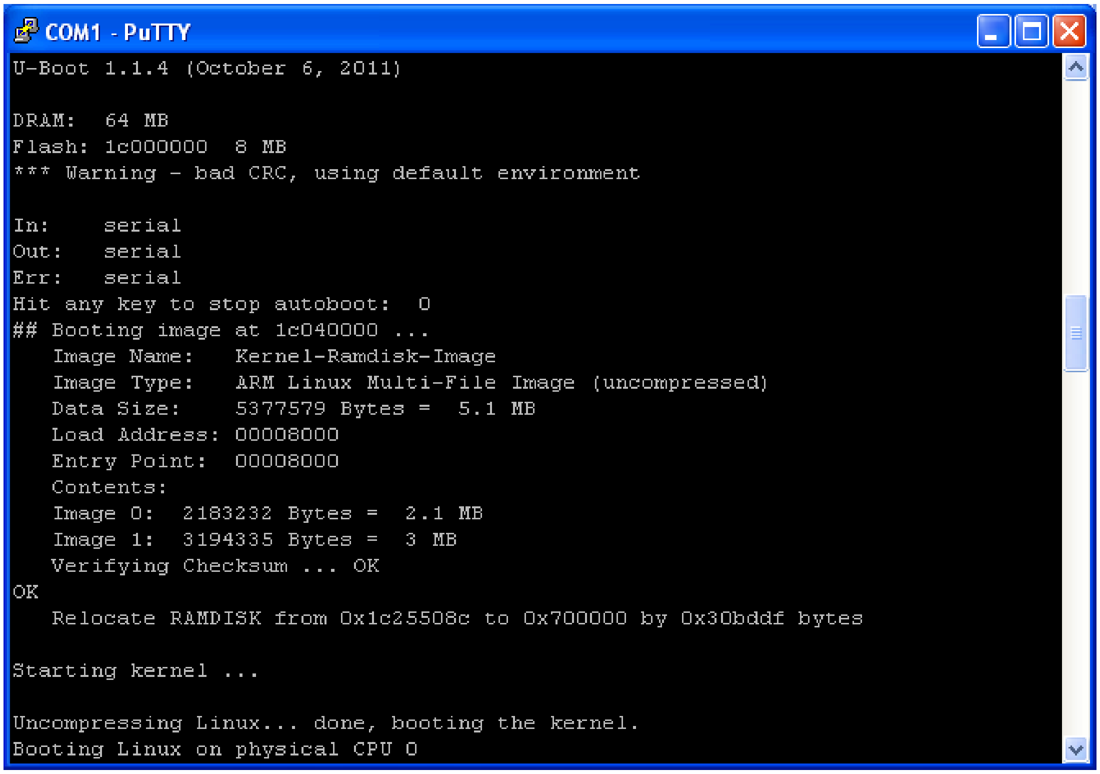
Figure 6 Location of 1PPS Signal on GPIO6 on KSZ8441 Board

4.5 Software Setup

The SOC board comes loaded with a Linux 2.6 environment, multiple PTP stacks, an AVB stack, and two utilities. This is the engine that executes the PTP protocol across a network utilizing Micrel KSZ84xx based devices.

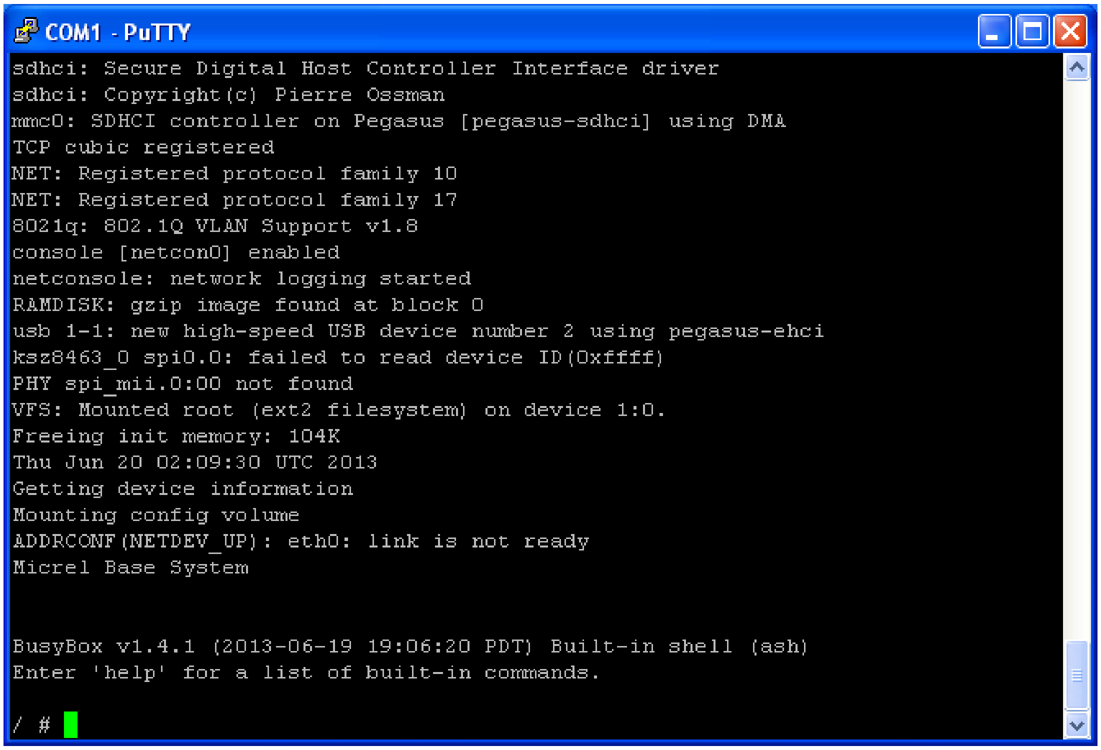
There are two areas of code that are loaded onto the SOC board at time of programming; the Uboot image and the Linux image. The Uboot area contains code that sets ups all the communication paths, interrogates the SOC device and communicates the available memory resources to the Linux OS.

After power up, when the SOC board and the PC are communicating, you should see activity as indicated in Figure 7**Error! Reference source not found.**, with some final communication ending with the Linux OS prompt being displayed in the command line window as shown in Figure 8. Note that the startup output from the SOC begins as soon as the board is powered up. If the communication program is not already invoked at power up, you will miss seeing the output shown in Figures 7 and 8. In this case, startup can be initiated by pressing the reset button on the SOC board.



```
U-Boot 1.1.4 (October 6, 2011)

DRAM:  64 MB
Flash: 1c000000  8 MB
*** Warning - bad CRC, using default environment

In:    serial
Out:   serial
Err:   serial
Hit any key to stop autoboot:  0
## Booting image at 1c040000 ...
   Image Name:   Kernel-Ramdisk-Image
   Image Type:   ARM Linux Multi-File Image (uncompressed)
   Data Size:    5377579 Bytes =  5.1 MB
   Load Address: 00008000
   Entry Point:  00008000
   Contents:
   Image 0:  2183232 Bytes =  2.1 MB
   Image 1:  3194335 Bytes =  3 MB
   Verifying Checksum ... OK
OK
   Relocate RAMDISK from 0x1c25508c to 0x700000 by 0x30bddf bytes

Starting kernel ...

Uncompressing Linux... done, booting the kernel.
Booting Linux on physical CPU 0
```

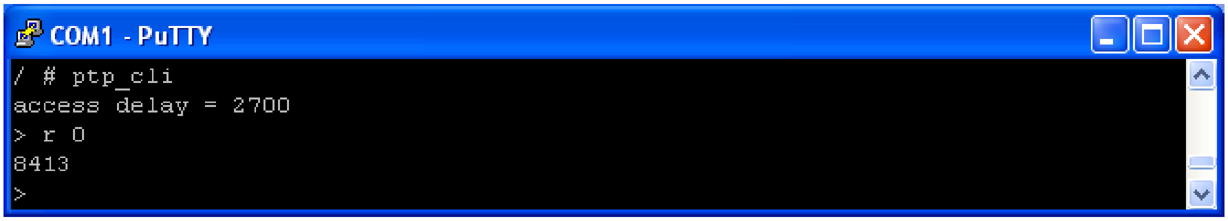
Figure 7 Startup of Communication Activity



```
sdhci: Secure Digital Host Controller Interface driver
sdhci: Copyright(c) Pierre Ossman
mmc0: SDHCI controller on Pegasus [pegasus-sdhci] using DMA
TCP cubic registered
NET: Registered protocol family 10
NET: Registered protocol family 17
8021q: 802.1Q VLAN Support v1.8
console [netcon0] enabled
netconsole: network logging started
RAMDISK: gzip image found at block 0
usb 1-1: new high-speed USB device number 2 using pegasus-ehci
ksz8463_0 spi0.0: failed to read device ID(0xffff)
PHY spi_mii.0:00 not found
VFS: Mounted root (ext2 filesystem) on device 1:0.
Freeing init memory: 104K
Thu Jun 20 02:09:30 UTC 2013
Getting device information
Mounting config volume
ADDRCONF(NETDEV_UP): eth0: link is not ready
Micrel Base System


BusyBox v1.4.1 (2013-06-19 19:06:20 PDT) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

/ #
```

Figure 8 Linux Prompt - Communication OK

To test that the SOC board is communicating properly with the KSZ8441, invoke the ***ptp_cli*** utility and read register 0. You should see the output shown in Figure 9.



```
/ # ptp_cli
access delay = 2700
> r 0
8413
>
```

Figure 9 Testing SOC to KSZ8441 Communication

The production version of the KSZ8441 returns value 8413, while the pre-production version returns value 8411. Any other value or message indicates a communication problem between the two boards. The "access delay" parameter is unimportant and may be ignored.

4.5.1 IP and MAC Addresses

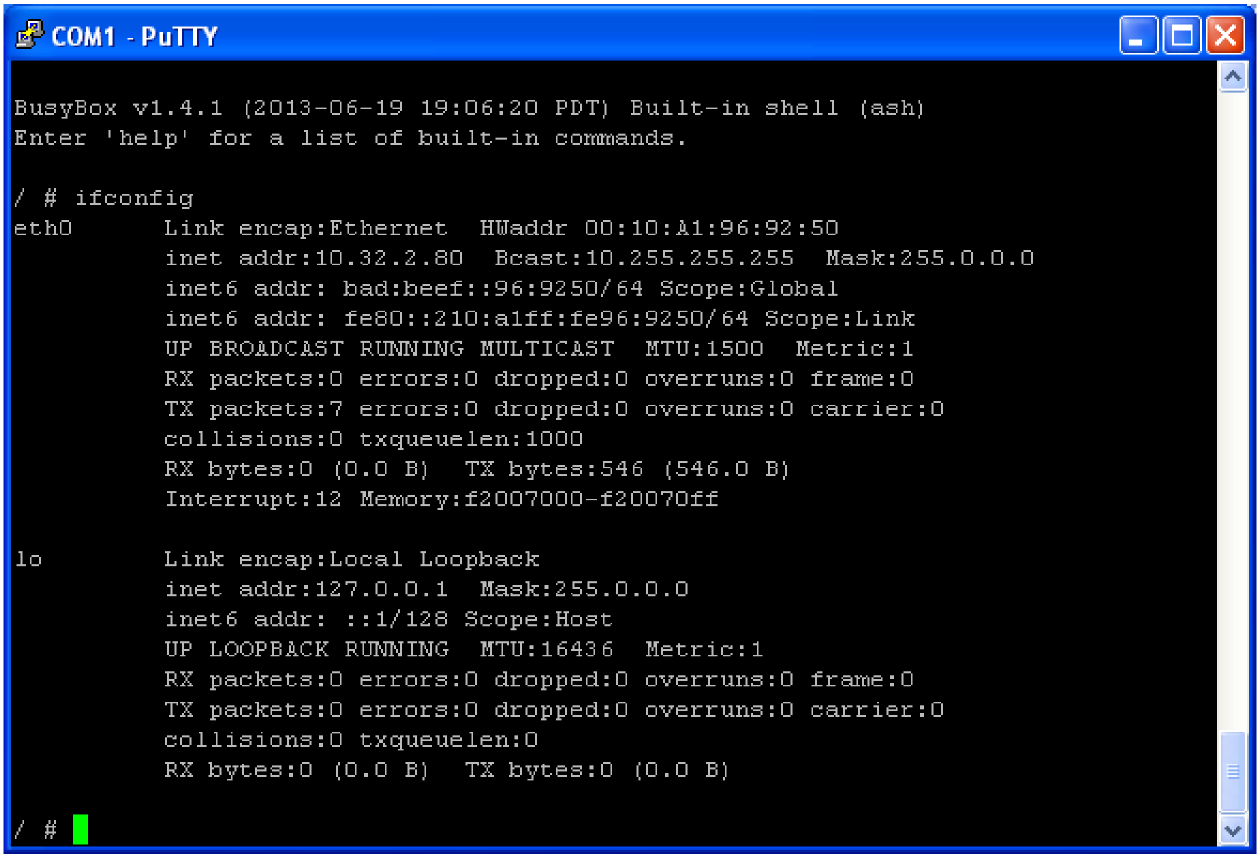
In a network, each SOC board should have its own unique IP and MAC address. When setting up the network, make sure to check that each one has a different address. SOC boards are typically shipped with the same IP and MAC address, so multiple SOC boards must have their addresses changed before being connected into a network.

If the addresses of an SOC board are unknown, they can be determined by using the serial port connection method described earlier, and typing the command ***ifconfig***, as shown in Figure 10. This returns information for the two ports on the SOC: **`eth0`** and **`lo`**. Port **`lo`** can be ignored. **`eth0`** is the relevant port. Figure 10 shows the default addresses:

MAC:	`00:10:A1:96:92:50`
IP:	`10.32.2.80`
IP mask:	`255.0.0.0`

Note that earlier revisions of the software had different defaults for the MAC and IP addresses.

When communicating (Telnet) between the PC and an SOC board, they must have compatible IP addresses, as determined by their IP masks. For the default values shown here, the PC must have an IP address of 10.xxx.xxx.xxx, where "xxx" is a value from 0 to 255. However, if the PC has a more restrictive IP mask, the PC address may be required to be 10.32.xxx.xxx or 10.32.2.xxx.



```
BusyBox v1.4.1 (2013-06-19 19:06:20 PDT) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

/ # ifconfig
eth0      Link encap:Ethernet  HWaddr 00:10:A1:96:92:50
          inet addr:10.32.2.80  Bcast:10.255.255.255  Mask:255.0.0.0
          inet6 addr: bad:beef::96:9250/64 Scope:Global
          inet6 addr: fe80::210:a1ff:fe96:9250/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:7 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:546 (546.0 B)
          Interrupt:12 Memory:f2007000-f20070ff

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

/ #
```

Figure 10 Linux ifconfig Command

The IP and MAC addresses are stored in the onboard software as parameters, and can be modified as described in the next section.

4.5.2 Modifying the Parameters Files

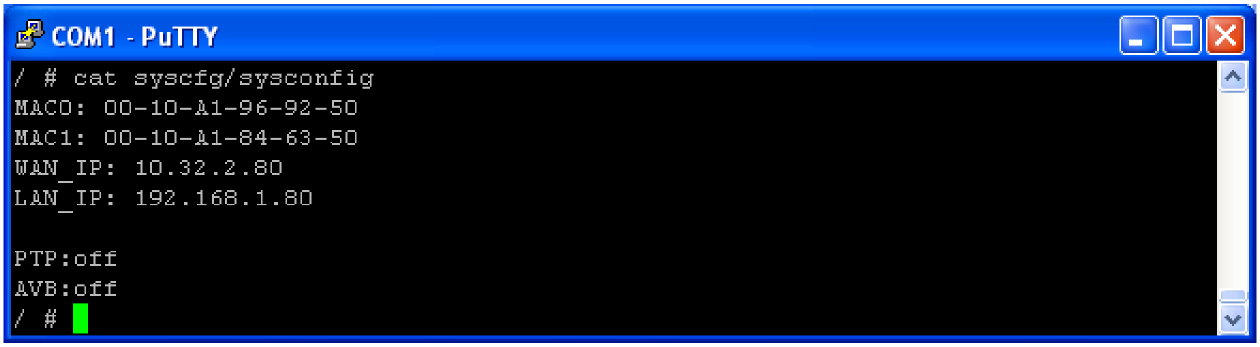
There are two text files in the environment that contain parameters which modify the operation of the SOC and KSZ8441 boards: ***sysconfig*** and ***ptpProfile.txt***. Both files are initially located in the **/etc** directory, but must be copied to the **/syscfg** directory before editing.

sysconfig

- MAC and IP addresses
- PTP and AVB on/off (auto-start)

The parameters stored in ***sysconfig*** are referenced by Linux during startup. The MAC address is specified in parameter ***MAC0***, and the IP address is specified in parameter ***WAN_IP***. The other parameters, MAC1 and LAN_IP, are not used for KSZ8441 evaluation. The parameters ***PTP*** and ***AVB*** can have the values "**off**" or "**on**". **On** indicates that the corresponding default PTP or AVB stack will start automatically when Linux starts up. **Off** indicates that it will not. If both parameters are set to **on**, then only the AVB stack will run.

The default contents of ***sysconfig*** are shown in Figure 11.



```
/ # cat syscfg/sysconfig
MAC0: 00-10-A1-96-92-50
MAC1: 00-10-A1-84-63-50
WAN_IP: 10.32.2.80
LAN_IP: 192.168.1.80

PTP:off
AVB:off
/ #
```

Figure 11 Sysconfig parameter file

ptpProfile.txt

- Slave only = 0/1
- End-to-End (E2E) or Peer-to-Peer (P2P)
- Sync message rate
- Announce message rate
- Priority1 and Priority2
- 1 step or 2 step operation
- Network protocol (UDP/IPv4, UDP/IPv6 or IEEE 802.3)

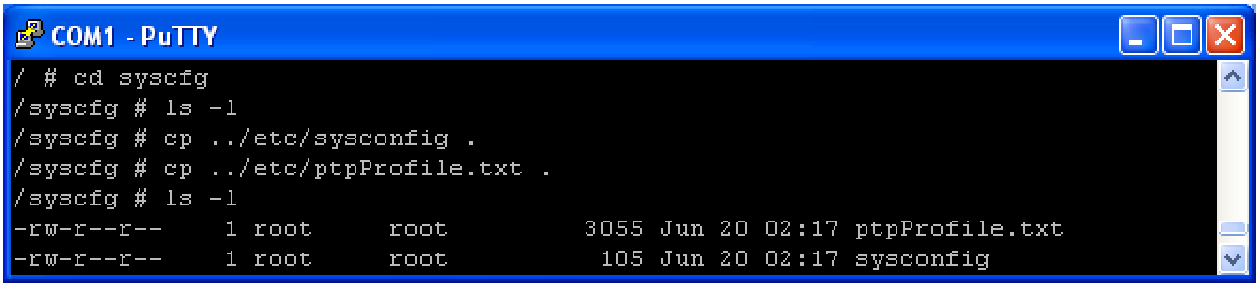
The parameters stored in ***ptpProfile.txt*** are used by the OnTime Networks PTP software stack, which is the default PTP stack described in this document. It is not used by the other PTP stacks that are included in the KSZ9692 PTP software.

The files ***sysconfig*** and ***ptpProfile.txt*** are initially located in the directory ***/etc***, but edits made to the files in this directory will have no effect.

To make changes to either ***sysconfig*** or ***ptpProfile.txt***, follow these steps:

1. Copy the file from the **/etc** directory to the **/syscfg** directory.
2. Use the text editor Vi to edit the file in the **/syscfg** directory.
3. Reset the SOC board or type the Linux command **reboot**.

The file copying process is shown in Figure 12.



```
/ # cd syscfg
/syscfg # ls -l
/syscfg # cp ../etc/sysconfig .
/syscfg # cp ../etc/ptpProfile.txt .
/syscfg # ls -l
-rw-r--r--    1 root     root         3055 Jun 20 02:17 ptpProfile.txt
-rw-r--r--    1 root     root          105 Jun 20 02:17 sysconfig
```

Figure 12 Copying files to syscfg directory

4.5.3 PTP Utility Programs

There are two utilities included with the SOC onboard software. One is ***ptp_cli*** and the other is ***ptp.exe***. These utilities allow the user to interact and explore the functionality of the KSZ84xx PTP environment.

4.5.3.1 ptp_cli Utility

The ptp_cli utility runs in the Linux environment and comes shipped installed on the SOC board. This is a command line utility that can be used to control the PTP hardware. This utility can read and write all registers within the KSZ8441. As such, it can also be used to set up GPIO pins to generate user defined waveforms with respect to the PTP clock.

It is invoked at the Linux command line prompt "`/ #`" by typing the following command. No parameters are used when calling ***ptp_cli***.

Refer to the Micrel PTP Utilities User Guide for detailed information.

4.5.3.2 ptp Utility

The SOC board comes installed with the ***ptp*** utility in the onboard Linux environment. A version is also available to run in the Windows environment: ***ptp.exe***.

The ***ptp*** utility provides a mechanism for sending PTP messages for testing and development of the PTP platform. Refer to the Micrel PTP Utilities User Guide for detailed information.

5 Using the KSZ8441HL Evaluation Platform

The Evaluation Kit is intended to provide a platform that enables designers to investigate and evaluate the capabilities of the KSZ8441 device. It is not intended to be a complete development system to be used for an entire product design effort.

While much work and development can be accomplished by using any of the setups described earlier, the setup described in Figure 2 offers a more realistic and robust environment in which to work. There are other setups and topologies that can be evaluated and is limited only by the imagination of the user. This section offers a few suggestions of activities that can be undertaken with the use of this Evaluation Kit.

Investigation of Various PTP Topology Parameters

- Point to Point topology
- End to End Topology
- One Step PTP operation
- Two Step PTP operation
- Selection between Multiple masters (best master algorithm)

GPIO/Input-Timestamp/Output Triggering

- Creation of register settings to generate custom waveforms on the GPIO outputs
- Creation of register setting to intercept and recognize waveforms at the GPIO inputs

PTP Protocol Execution Under Varying Network Traffic Patterns

- Introducing the desired network traffic into the device

5.1 Executing the PTP Stack

The original software revision included a single PTP stack from OnTime Networks. Later revisions are including multiple PTP stacks, as well as an AVB stack. This document only describes the use of the

OnTime stack. For details about using the other stacks, refer to the ***README*** file located in the **/ptp** directory.

Note: Early revisions of the PTP stacks may not be fully compatible with the final production version of the KSZ8441HL. Users who have a Linux image dated prior to year 2013 should update to the latest revision if they KSZ8441HL silicon dated newer than 2012.

After power-up has successfully completed, the system will be sitting idle with the Linux prompt displayed as in Figure 8. The system is now ready to accept your commands. To invoke the PTP stack, type in the following at the Linux prompt:

```
etc/ptpinit –d
```

The script ***ptpinit*** runs the OnTime Networks PTP protocol stack, which utilizes the parameters in the file ***ptpProfile.txt***.

The "-d" command option sets verbose mode, thereby displaying ongoing status information to the screen.

After typing in the command and hitting return, the screen will display various lines of text as the PTP stack communicates across the network to establish a system that has all nodes in sync. During this time, you will see various statements with the words "CALIB1", "CALIB2", 'CALIB3", "CALIB4", and "CALIB5" displayed. They will be displayed in numerical order as the process progresses. During this time, the system is establishing synchronization. When synchronization is successful, a final message will be displayed which has the words "MASTER_SELECTED". From that point on, the various statistics and PTP messages and information will be displayed periodically.

<ctrl> c may be used to stop the PTP stack.

6 Frequently Asked Questions and Help

Hopefully you will find the answers to your questions here before needing to contact Micrel for assistance.

6.1 Frequently Asked Questions

Q1. How many KSZ84xx Evaluation Boards can I put into the system to test out the PTP environment?

A1. The software currently has a limitation of 64 nodes for the Master to keep track of. Any value over that may cause issues depending on the nature of the traffic.

Q2. If the system hangs and stops functioning, what is the best recovery procedure to follow?

A2. There is a reset switch on both boards. Reset the boards to attempt recovery.

Q3. How can I change the operating environment from "End to End" topology to "Peer to Peer" topology?

A3. Make changes to the ptpProfile.txt file.

Q4. Can I use other criteria for determining the choice of the Grandmaster source?

A4. Yes, the Priority values in the ptpProfile.txt file can be changed. A lower value on Priority1 or Priority2 makes the node a higher priority.

Q5. How do I know if SOC board firmware updates are available?

A5. Please check the KSZ8441 page on the Micrel website.

Q6. How do I update the onboard SOC firmware to the latest revision available from Micrel?

A6. There are two sections of the onboard firmware; the U-Boot section that contains the Linux OS which rarely gets modified and the Micrel unique code area which gets updated periodically as needed. Micrel has a document available which goes over the details of updating either section. A third party JTAG programming module may be used. There is also an alternative method of updating the firmware via the Ethernet Port or Serial Port. Contact Micrel for an application note that describes the steps for programming the SOC firmware.

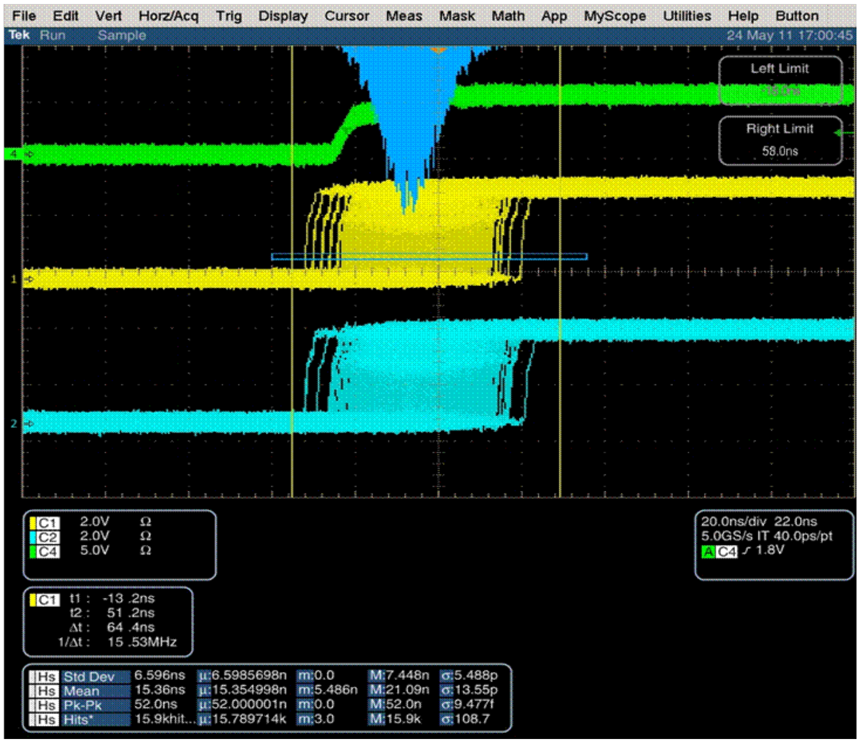
Q7. How should the oscilloscope be set up to properly monitor the 1 PPS signals from the KSZ8441 Evaluation boards and the Grandmaster or Master source?

A7. Typically, the 1 PPS signal from the Grandmaster or Master node is used as the trigger on channel one of the oscilloscope. Channel 2 is connected to the 1 PPS signal on one of the other KSZ8441 Evaluation boards. The 1 PPS signal is generated on the GPIO6 signal pin. If the oscilloscope has a persistence feature, it can be used to view accumulating past occurrences of the signal on channel two with respect to the 1 PPS signal on channel 1. Depending on the oscilloscope used, there is other statistical analysis that can be performed on the signals. Figure 13 illustrates an oscilloscope signal capture.

Master Clock (MC)
1 PPS reference

Transparent Clock (TC1)
Jitter = 52 ns
Std deviation = 6.6 ns

Transparent Clock (TC2)



```
Right Limit 58.0ns

C1 2.0V Ω
C2 2.0V Ω
C4 5.0V Ω

20.0ns/div  22.0ns
5.0GS/s IT 40.0ps/pt
A C4 ⨍ 1.8V

C1  t1 :   -13 .2ns
    t2 :    51 .2ns
    Δt :    64 .4ns
  1/Δt :    15 .53MHz

Hs Std Dev  6.596ns    µ:6.5985698n  m:0.0     M:7.448n  σ:5.488p
Hs Mean     15.36ns    µ:15.354998n  m:5.486n  M:21.09n  σ:13.55p
Hs Pk-Pk    52.0ns     µ:52.000001n  m:0.0     M:52.0n   σ:9.477f
Hs Hits*    15.9khit…  µ:15.789714k  m:3.0     M:15.9k   σ:108.7
```

Figure 13 Sample 1 PPS Oscilloscope Capture

Q8. What does it mean if I do not see the “MASTER_SELECTED” message on PTP startup?

A8. The PTP firmware was not successful in synchronizing all the nodes on the network. There is some problem in one of the nodes. One possible cause is that an old revision of the SOC software is interfacing to a newer revision of the KSZ8441HL. Check for the latest SOC software and consider updating your SOC board.

6.2 Help!

For assistance, contact your Micrel representative at the following location.

http://www.micrel.com/help_Info.jsp

7 Reference Material

KSZ8441HL Datasheet (Contact Micrel for latest datasheet)
IEEE 802.3 Specification
IEEE 1588 Specification

8 Revision History

Revision	Date	Summary of Changes
0.1	07/15/12	- Initial Release, used KSZ8462 06/19/11 revision.
0.2	8/15/13	- Extensive revision - Additional setup examples - Added quick start guide - Additional pictures

The information furnished by Micrel in this document is believed to be accurate and reliable. However, no responsibility is assumed by Micrel for its use. Micrel reserves the right to change circuitry and specifications at any time without notification to the customer. Micrel Products are not designed or authorized for use as components in life support appliances, devices or systems where malfunction of a product can reasonably be expected to result in personal injury. Life support devices or systems are devices or systems that (a) are intended for surgical implant into the body or (b) support or sustain life, and whose failure to perform can be reasonably expected to result in a significant injury to the user. A Purchaser's use or sale of Micrel Products for use in life support appliances, devices or systems is at Purchaser's own risk and Purchaser agrees to fully indemnify Micrel for any damages resulting from such use or sale.