User's Guide

DP83TC812 Troubleshooting Guide

Eric Tian, David Creger

Table of Contents

Trademarks

1 Preface

The DP83TC812-Q1 device is an IEEE 802.3bw compliant automotive Ethernet physical layer transceiver which operates with Single Twisted Pair cable. The PHY supports TC10 sleep and wake features and provides all physical layer functions needed to transmit and receive data over unshielded or shielded single twisted-pair cables. The device provides xMII flexibility with support for standard MII, RMII, RGMII, and SGMII MAC interfaces. The PHY also integrates a low pass filter on the MDI side to reduce emissions.

Figure 1-1 shows a high-level system block diagram of a typical DP83TC812 application.

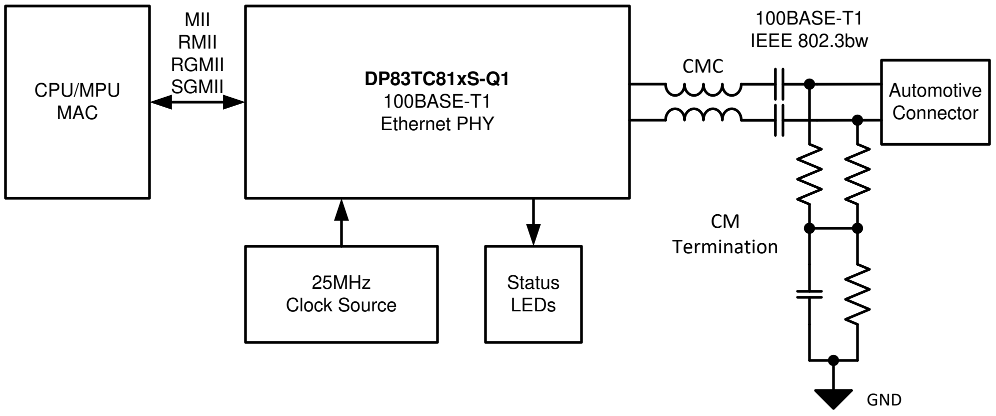


Figure 1-1. DP83TC812 Block Diagram

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers can be shown with the suffix h or the prefix 0x. For example, the following number is 40 hexadecimal (decimal 64): 40h or 0x40.
- Registers in this document are shown in figures and described in tables.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties with default reset value below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure can have one of multiple meanings:
    - Not implemented on the device
    - Reserved for future device expansion
    - Reserved for TI testing
    - Reserved configurations of the device that are not supported
  - Writing nondefault values to the Reserved bits could cause unexpected behavior and should be avoided.

2 Related Documentation

- Texas Instruments, *DP83TC812, DP83TC813, and DP83TC814: Configuring for Open Alliance Specification Compliance* application note.
- Texas Instruments, *DP83TC812, DP83TC813: System Implementation of Open Alliance TC10 Sleep/Wake-up* application note.
- Texas Instruments, *DP83TC812-USB-2-MDIO-SCRIPTS* tool.
- Texas Instruments, DP83TC812-SCHEMATIC-REVIEW-CHECKLIST tool.

3 Support Resources

TI E2E™ support forums are an engineer's go-to source for fast, verified answers and design help — straight from the experts. Search existing answers or ask your own question to get the quick design help you need.

Linked content is provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

4 Troubleshooting the PHY Application

This guide was intended to help diagnose and resolve a variety of common PHY issues such as link up not achieved, register read/write issues, throughput issues, etc.

The following sections approach the debug by starting with items with broad impact and proceeding to progressively finer focused items. We recommend to follow the flow of this document in the order listed but some sections can be passed over depending on the nature of the debug.

4.1 Schematic and Layout Checklist

Schematic and or layout design issues can have a broad impact on the performance of the device, leading to issues from a totally non-functional PHY to a few bit errors in the stream. The following spreadsheet contains a pin-by-pin checklist to verify the schematic and layout design meet TI recommendations. A strap tool is available to maintain that the correct bootstrap modes have been enabled, which is another common issue. Use the drop-down on the blue cells to select your specific configuration and complete the pin-wise checklist and strap tool below.

DP83TC812 Schematic and Layout Checklist

4.2 Verify Successful Power-up of PHY

The first obvious but often overlooked item is proper power being supplied to the PHY. Probe each of the four supply rails of the PHY to maintain that the voltages are within limits shown below.

Table 4-1. PHY Supply Voltage Specifications

	Pin Number	Descriptions	Min	Typ	Max	Unit
VDDIO, VDDMAC	34, 22	IO Supply Voltage = 1.8V	1.62	1.8	1.98	V
		IO Supply Voltage = 2.5V	2.25	2.5	2.75	
		IO Supply Voltage = 3.3V	2.97	3.3	3.63	
VDDA	11	Core Supply Voltage	2.97	3.3	3.63	V
VSLEEP	7	Sleep Supply Voltage	2.97	3.3	3.63	V

4.3 Peripheral Pin Checks

The following section details the expected values of various peripheral output pins of the PHY during operation. Measure and compare the noted pin outputs to verify PHY operation.

4.3.1 Probe the RESET_N pin

The reset input (pin 3) is active low. Confirm that the controller is not driving the RESET_N signal low. Otherwise, the device is held in reset and appears non-functional and register access is not available. Make sure the RESET_N pin is equal to VDDIO voltage.

4.3.2 Probe the INH pin

The INH signal (pin 10) gives an indication if the device is in sleep mode or not. This pin is Hi-Z when the PHY is in TC-10 SLEEP and pulled low by external pull resistor. This pin is driven high for all other PHY states. In sleep mode, the device appears non-functional and register access is not available. Make sure this pin is equal to Vsleep voltage to be sure the device is not in sleep mode.

4.3.3 Probe the CLKOUT pin

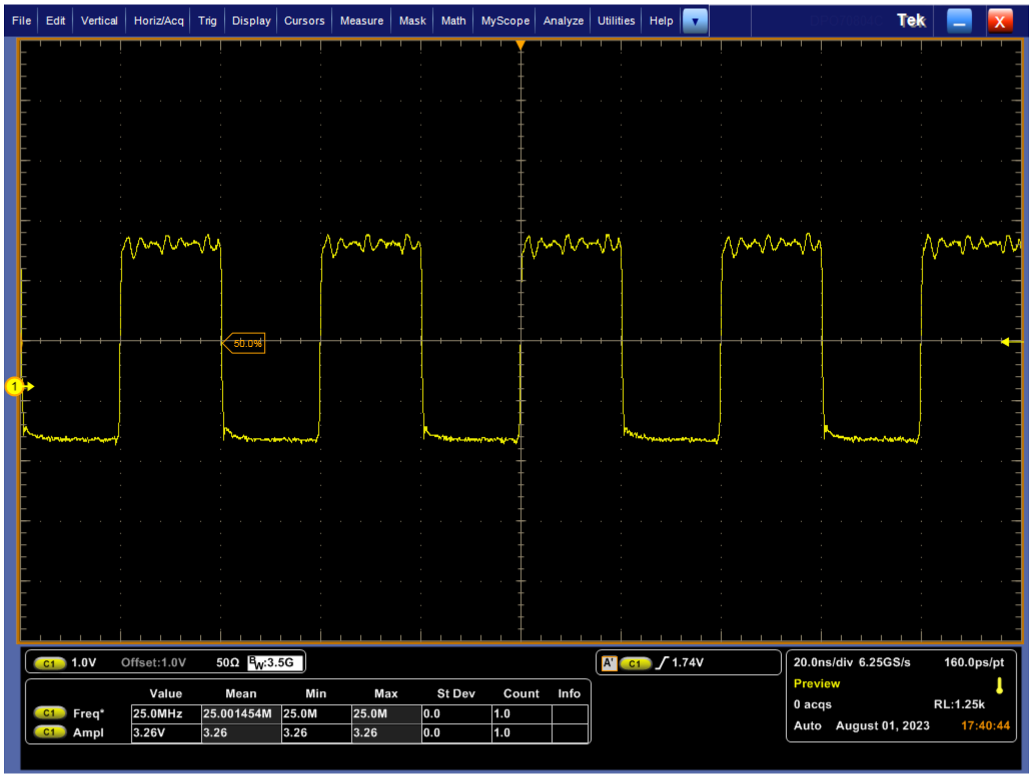
The reference clock frequency and stability is of utmost importance to maintain proper operation of the PHY. Failure to meet the specifications in the data sheet can lead to bit errors, read/write issues, or total non-operation of the PHY.

Do not probe the crystal directly as this can change the capacitive loading of the circuit and alter the behavior. Instead, probe the CLKOUT pin (pin 16) which is a buffered version of the input reference clock.

Maintain that the frequency is within ±100ppm of the expected value: (40.995 - 50.005Mhz) for RMII slave mode and (24.9975 - 25.0025Mhz) for all other modes.

Table 4-2. 25MHz Crystal Requirements

25MHz Crystal Requirements		
Frequency	25	MHz
Max Frequency tolerance and Stability over Temperature and Aging	±100	ppm
Max Equivalent Series Resistance	100	Ω



	Value	Mean	Min	Max	St Dev	Count	Info
C1 Freq*	25.0MHz	25.001454M	25.0M	25.0M	0.0	1.0	
C1 Ampl	3.26V	3.26	3.26	3.26	0.0	1.0	

Figure 4-1. DP83TC812 CLK_OUT Measurements

Note

Verify that the amplitude is equal to VDDMAC and Frequency is within ±100ppm (24.9975 - 25.0025Mhz)

4.3.4 Probe the Serial Management Interface (MDC, MDIO) Pins

If register read and write is successful, this section can be skipped.

If register read and write is unsuccessful, probe the MDC signal (pin 1) to maintain that there is a ≤20Mhz clock signal being sourced from the MAC. Additionally, the MDIO signal (pin 36) can be decoded using a logic analyzer as shown below.

Note, to access extended registers (those beyond 0x1F), the procedure given in section 8.4.15 of the data sheet must be used.

Table 4-3. SMI Protocol Structure

SMI Protocol	**<idle><start><op code><device addr><reg addr><turnaround><data><idle>**
Read Register	<idle><01><10><AAAAA><RRRRR><Z0><XXXX XXXX XXXX XXXX><idle>
Write Register	<idle><01><01><AAAAA><RRRRR><10><XXXX XXXX XXXX XXXX><idle>

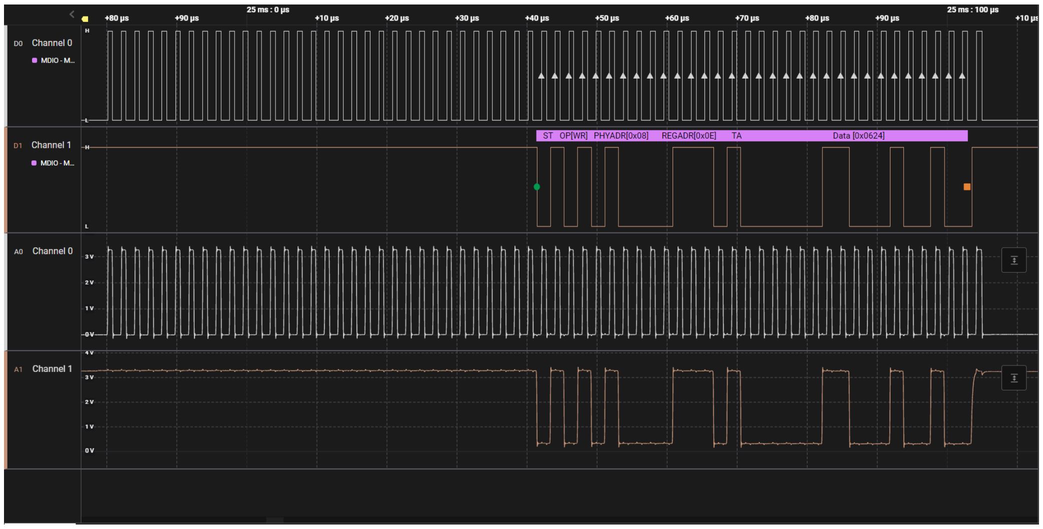


Figure 4-2. MDC/MDIO Write Example

Note

MDC/MDIO write operation for register 0xE = 0x0624 on PHY address 8.

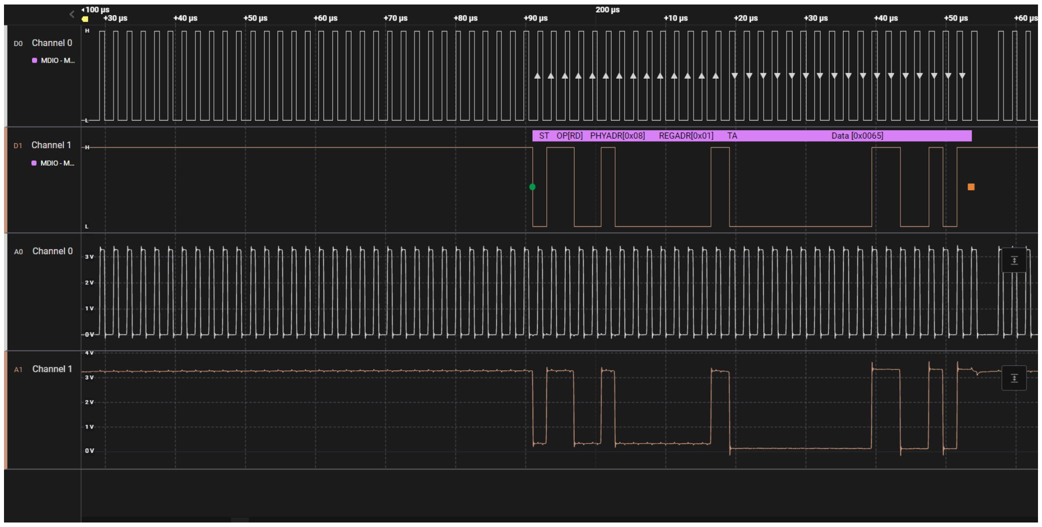


Figure 4-3. MDC/MDIO Read Example

Note

MDC/MDIO read operation on register 0x1 of PHY address 8.

4.4 Register Dump Comparison

Read each of the below registers and verify the values shown. Note that the initial values of some registers can vary based on strap options. Comparing a register dump to the one shown helps to highlight any values different from the expected.

The below register dump shows the expected values when link is up and the PHY is in RGMII mode, MDI slave, with PHY address 0xA.

Table 4-4. DP83TC812 Register Value Check

REGISTER ADDRESS	REGISTER NAME	REGISTER VALUE	DESCRIPTIONS
0x0000	BMCR	0x2100	
0x0001	BMSR	0x0065	Bit[2] shows that link is up
0x0002	PHYIDR1	0x2000	
0x0003	PHYIDR2	0xA271	0xA271 is the unique identifier for the DP83TC812 PHY. A value other than this indicates a different PHY is connected.
0x0010	PHYSTS	0x0005	
0x0011	PHYSCR	0x010B	
0x0012	MISR1	0xE400	Indicates the presence of any interrupts
0x0013	MISR2	0x0000	Indicates the presence of any interrupts
0x0015	RECR	0x0000	Receive error counter
0x0016	BISCR	0x0100	
0x0018	MISR3	0x5825	Indicates the presence of any interrupts
0x0019	REG_19	0x0C0A	Bits 4-0 is the decoded PHY address from strap
0x001B	TC10_ABORT_REG	0x0000	
0x001E	CDCR	0x0000	
0x018B(1)	LPS_CFG2	0x1C0B	Bit[6] indicates autonomous or managed mode. Note the PHY will not automatically link up if this bit is 0.

Table 4-4. DP83TC812 Register Value Check (continued)

REGISTER ADDRESS	REGISTER NAME	REGISTER VALUE	DESCRIPTIONS
0x045D[(1)]	CHIP_SOR_1	0x408C	Sampled PHY Strap configurations after power up or reset. Verify with strap tool in the schematic checklist for specific configurations.
0x0600[(1)]	RGMII_CTRL	0x0038	Bit[3] indicates that RGMII mode is enabled
0x0608[(1)]	SGMII_CTRL_1	0x007B	Bit[9] indicates that SGMII mode is disabled
0x0648[(1)]	RMII_CTRL_1	0x0120	Bit[6] indicates that RMII mode is disabled
0x1834[(1)]	MMD1_PMA_CTRL_2	0x8000	PHY Master/Slave configuration. Value will read 0xC000 in master mode and 0x8000 in slave mode

Note

Registers above 0x1F are extended registers and must be accessed using the extended register access procedure.

4.5 Verifying Strap Configurations

Incorrect strap configurations are one of the most common issues leading to lack of data throughput. For example, if the incorrect MAC interface mode is chosen or the PHY is placed in managed mode when autonomous mode is intended to be used, data transmission and reception is not successful.

The strap values sampled at power up can be read from register 0x45D (CHIP_SOR_1) using Extended Register Access. Use the strap tool in the DP83TC812 Schematic Checklist to verify the intended straps configuration has been loaded in register 0x45D.

Verify that the MAC is not driving any pins connected to the PHY upon power up or reset pin de-assertion. This causes an incorrect voltage to be sampled and cause the strap value to be different than intended. All MAC pins connected to the PHY must be placed in a High-Z state while the PHY is powering up or coming out of reset.

4.6 Check the MDI Signal

Check for the presence of a signal on the MDI to confirm the PHY is in master mode and the transmitter is operational. Note in slave mode the MDI is silent.

With link partner disconnected, set the PHY to MDI master and place a 100Ω resistor between the pins of the automotive connector (or use an automotive compliance fixture) and use a differential probe to measure the output signal on an oscilloscope. Shown below is the expected signal. In a later section, Section 4.12, we verify the transmitter characteristics are meeting Open Alliance compliance standards. For now, the presence of signal here indicates the PHY is receiving power, reference clock, and is not in sleep mode.

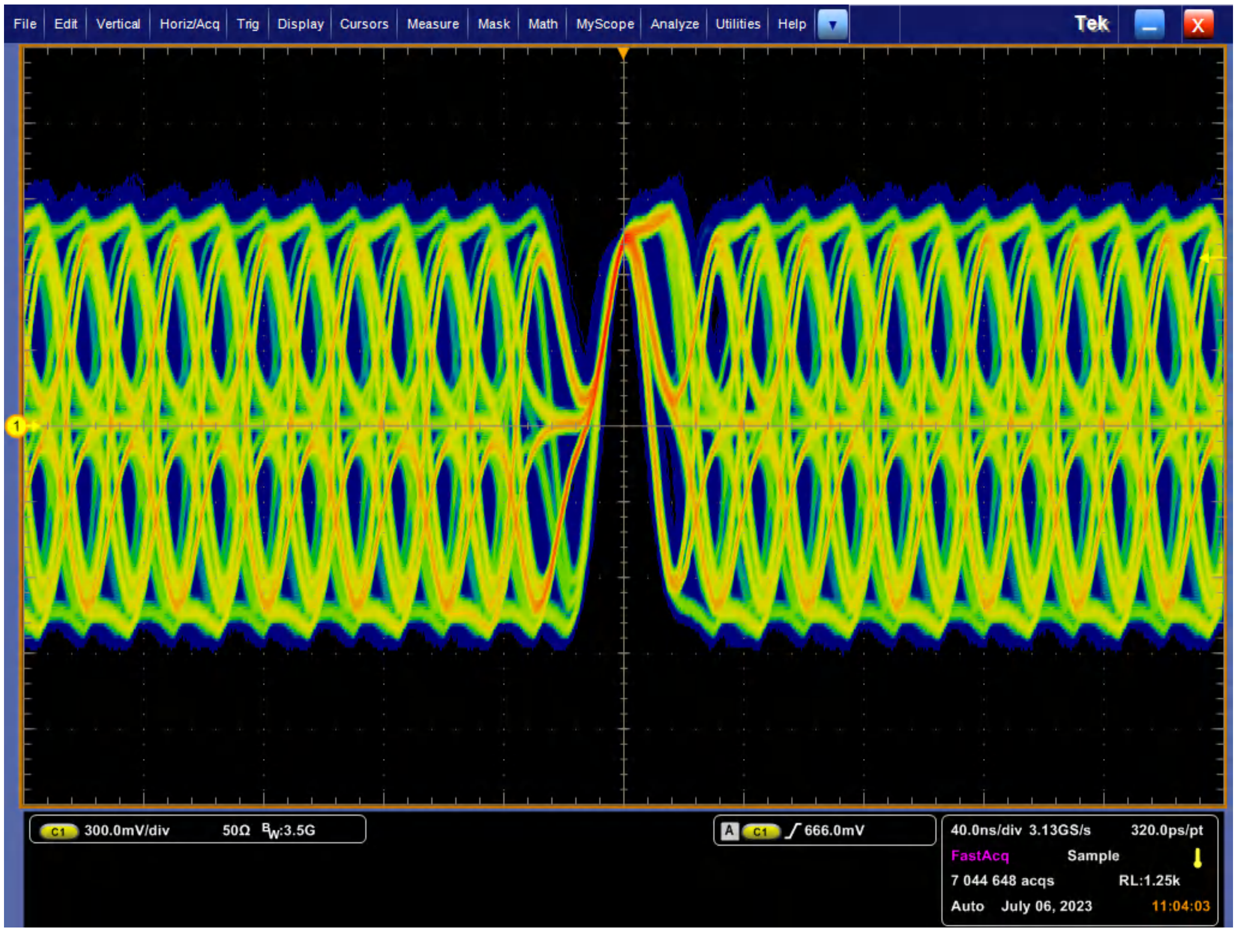


Figure 4-4. DP83TC812 Master Idle Symbol

4.7 Link Up Failed Common Issues

Below are some common culprits if link up cannot be achieved.

1. One link partner must be in MDI master mode while the other is in MDI slave mode. Verify this is the case by reading register 0x1834 on both PHYs.
2. Unless link up is intended to be initially blocked upon power up of PHY, verify the device is in autonomous mode by confirming register 0x18B bit[6] = 1.
3. Verify the PHY is not asleep by probing the INH pin and confirming the voltage is equal to Vsleep.
4. Connect a different automotive ethernet cable to confirm the cable length, type, or health is not preventing the link up.
5. Verify the script given in SNLA389 has been written to the PHY.

4.8 Signal Quality Check

Once link up is confirmed by reading register 0x1 = 0x0065, you can check the quality of the link by utilizing the Signal Quality Indicator (SQI) feature of the DP83TC812. Poor link quality caused by layout or cable imperfections has the potential to lead to packet errors in the bit stream or link drops.

SQI is a method of quantifying the signal quality by measuring the signal to noise ratio, specified by Open Alliance. The value can range from 0 to 7 with 7 corresponding to the best link quality. Register 0x871 bits[3:1] can be read to identify the 3 bit SQI value. Read this register and verify the value is greater than 4 to maintain excellent link quality. Values less than 4 can point to an imperfection in schematic/layout, cable, or being in a noisy environment. Please verify that your design is meeting the recommendations given in the DP83TC812 Schematic and Layout Checklist.

Additionally, verify the script given in SNLA389 has been written to the PHY. This script must be written to maintain the best signal quality.

Table 4-5. DP83TC812 SQI Mapping

Reg 0x871[3:1]	Open Alliance SQI Level	Link Quality
0x0	0 (Worst)	Poor or No Link
0x1	1	
0x2	2	
0x3	3	
0x4	4	Good or Excellent Link
0x5	5	
0x6	6	
0x7	7 (Best)	

4.9 Power Up Timing

Meeting the power up timing requirements given in the data sheet is imperative for correct operation of the PHY. Failure to meet the power up timing requirements can lead to an inability to achieve link, read and write issues, or a completely non-operational PHY. Given below are the power up and reset timing requirements for the DP83TC812. Measure each of the following items on an oscilloscope to verify that every item is met.

1. Supply ramps start from 0V. Supply ramps are continuous and smooth (no pedestal voltages).
2. Ramp time is greater than 0.2ms and less than 8ms.
3. Oscillator is stable within 10ms of power ramp. Measure CLKOUT pin on one channel and supply ramp on another.
4. MDC toggling does not occur within 10ms of supply ramp or within 1ms of reset pin de-assertion.

Table 4-6. Power Up Timing Specifications

PARAMETER		MIN	NOM	MAX	UNIT
T5.1	Supply ramp time: For all supplies	0.2		0.8	ms
T5.3	XTAL Startup / Settling: Powerup to XI good/stabilized		0.35		ms
T5.4	Oscillator stabilization time from power up			10	ms
T5.5	Post power-up to SMI ready: Post Power-up wait time required before MDC preamble can be sent for register access	10			ms
T5.6	Power-up to Strap latch-in			10	ms
T5.7	CLKOUT Startup/Settling: Powerup to CLKOUT good/stabilized			10	ms
T5.8	Power-up to idle stream			10	ms

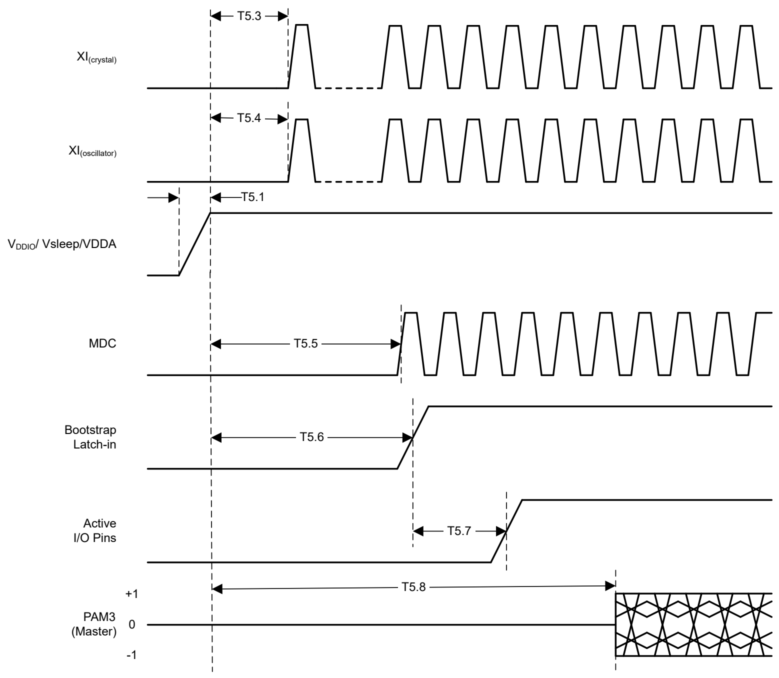


Figure 4-5. Power Up Timing Diagram

Table 4-7. Reset Timing Specifications

PARAMETER		MIN	NOM	MAX	UNIT
T6.1	Reset Pulse Width: Minimum Reset pulse width to be able to reset	720			us
T6.2	Reset to SMI ready: Post reset wait time required before MDC preamble can be sent for register access	1			ms
T6.3	Reset to Strap latch-in: Hardware configuration pins transition to output drivers		40		us
T6.4	Reset to idle stream			1800	us

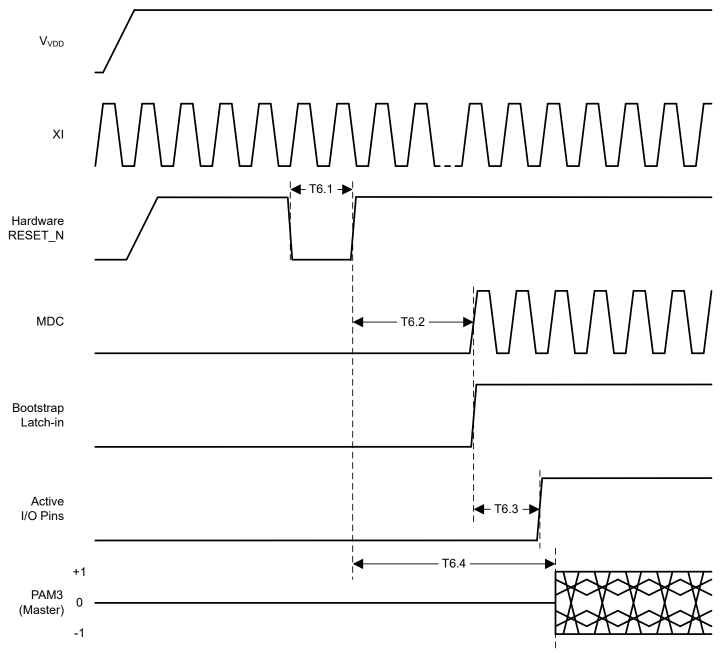


Figure 4-6. Reset Timing Diagram

4.10 Loopback Testing

When packets are not able to be received between nodes or bit errors are appearing in the stream, an integrated debugging tool called loopbacks can be used to narrow down the source of the problem. Loopback allows the PHY to transmit received data and isolate different parts of the data path to see where the problem resolves. In a connection between two nodes, a problem can lie on the MAC interface of PHY1, MAC interface of PHY2, or MDI connection between PHYs. Loopbacks can isolate which one of the three is the problem.

Two different kinds of loopback exist on the DP83TC812 PHY: MAC side (digital loopback) and cable side (reverse loopback). These can be used together to check each segment of the data path. Most MACs have packet generating and checking capabilities. A packet can be transmitted, looped back, and received by the MAC and compared to the original value. If the packet is not received or packet errors occur when a segment has been isolated from the data path, we know that to be the problem segment. Below shows the two tests required to isolate such a problem.

First, a MAC side loopback (digital loopback) is enabled on PHY 1 (by writing 0x16 = 0x0104). Packets are then generated and checked by MAC1. If errors are present, the problem is located on the xMII interface between MAC1 and PHY1. If not, the problem is elsewhere.

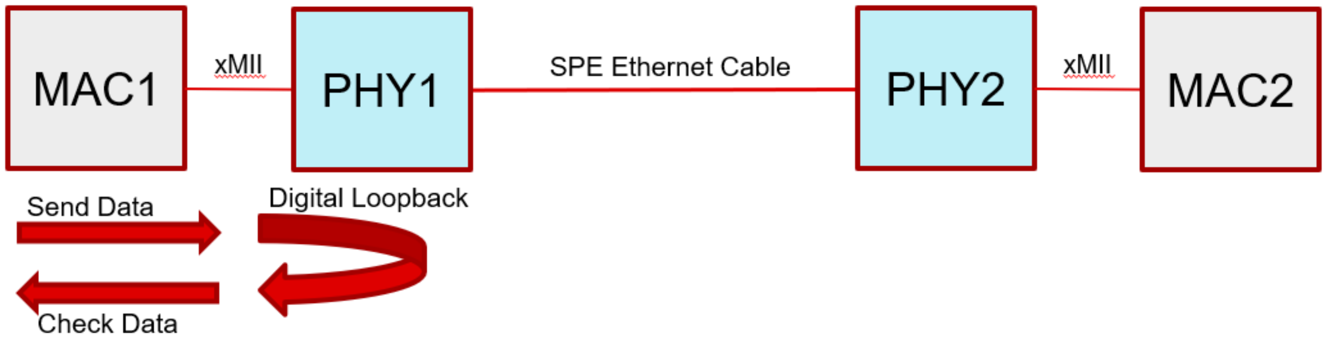


Figure 4-7. Utilizing Digital Loopback to Check Packets

Next, a cable side loopback is enabled on PHY2 (by writing 0x16 = 0x0110), and packets are once again generated and checked by MAC1. If the issue persists here but not in the previous step, the problem is located on the MDI interface between the two PHYs. If packet errors are still not present, the problem must be located on the xMII interface between PHY2 and MAC2.

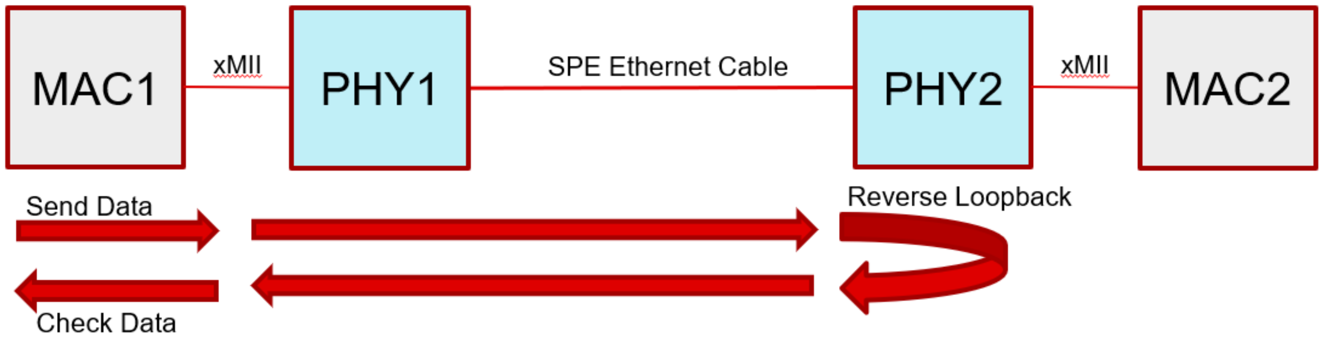


Figure 4-8. Utilizing Reverse Loopback to Check Packets

Now that we know where the problem lies, targeted investigation can be had at this location. For problems with the MDI, see Section 4.7. For problems with the xMII, see Section 4.11.

4.11 Debugging the MAC Interface

The most common cause of MAC interface communication failures is incorrectly setting the desired MAC interface mode. Link can be up but a ping command fails if, for example, SGMII mode is enabled but the PCB is designed for RGMII communication. Please see the strapping tool of the DP83TC812 Schematic Checklist to confirm the correct MAC interface mode is selected. MAC mode can also be selected by writing to registers 0x600[3], 0x608[9], 0x648[6].

Reduced Gigabit Media Independent Interface (RGMII)

The signals used for the RGMII protocol are shown below:

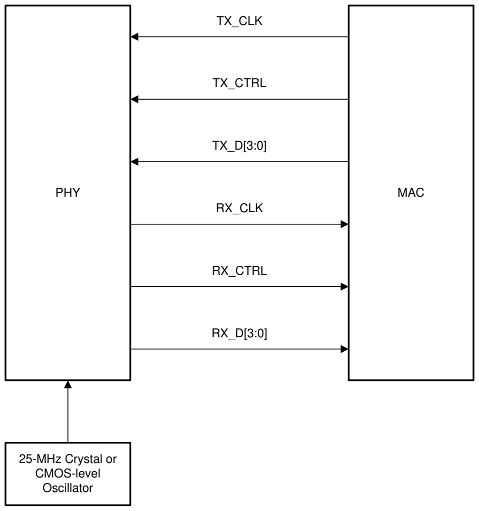


Figure 4-9. RGMII Signaling

The RGMII protocol has certain timing constraints which must be met for data to be properly received. These timing constraints are shown below in Table 4-9. Namely, a minimum of 1ns setup and 1ns hold time must be maintained at the input of the receiver. To meet this requirement, some amount of skew must be introduced between the clock and data signals. This skew can be introduced either by the MAC, the PHY, or as part of the PCB trace lengths. The DP83TC812 has two modes, align and shift mode, for both RX and TX signals. These modes are selected by bootstrap or can be adjusted in register 0x602. Note if shift mode is enabled on the PHY TX signals, the MAC transmits the data without skew. Similarly, if align mode is chosen on the PHY RX signals, the MAC must be set to RX shift mode. Table 4-8 shows the correct MAC and PHY RGMII delay configurations.

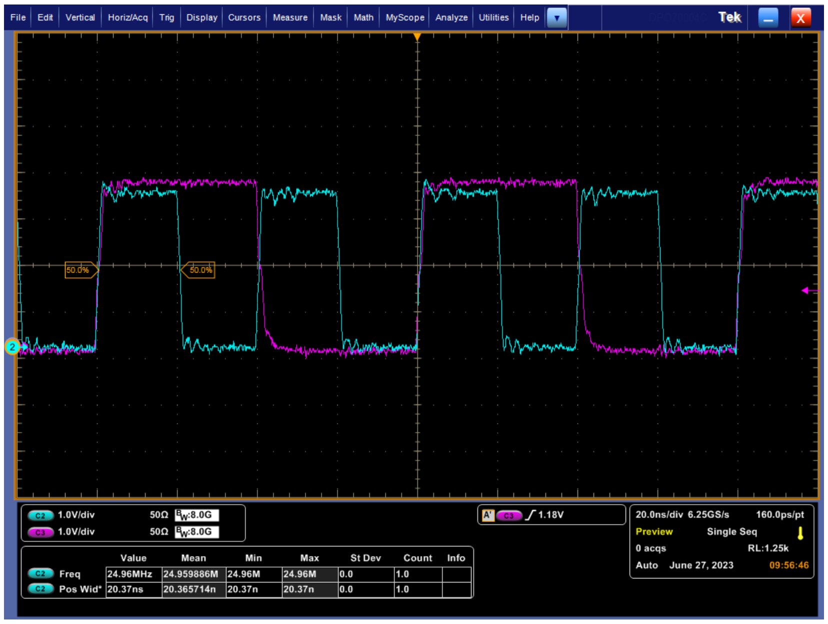
Table 4-8. RGMII Shift Configurations

MAC Configuration	Required PHY Configuration
RGMII Align on Rx	RGMII Shift on Rx
RGMII Shift on Rx	RGMII Align on Rx
RGMII Align on Tx	RGMII Shift on Tx
RGMII Shift on Tx	RGMII Align on Tx

In RGMII RX shift mode, the PHY shifts RX_CLK ahead of RX_Data signals by roughly 3ns.

When using the PHY's TX shift mode, the PHY is expecting TX_CLK and TX_Data signals to be aligned at its pins, and the data is shifted internally.

Reference waveforms below show the effect of enabling shift or align mode on RX_D0 signal.



C2	1.0V/div	50Ω	BW:8.0G
C3	1.0V/div	50Ω	BW:8.0G

A' C3 ∫ 1.18V

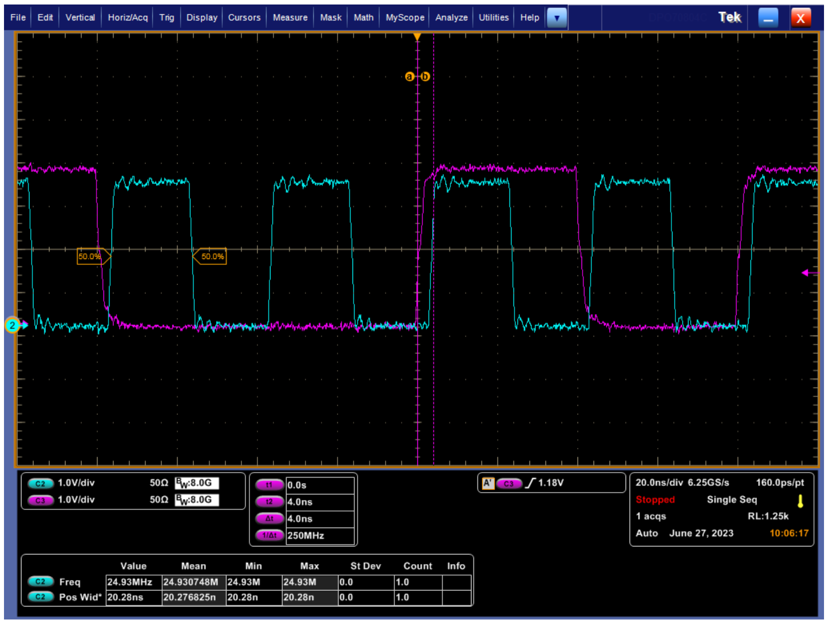
20.0ns/div 6.25GS/s 160.0ps/pt
Preview Single Seq
0 acqs RL:1.25k
Auto June 27, 2023 09:56:46

	Value	Mean	Min	Max	St Dev	Count	Info
C2 Freq	24.96MHz	24.959886M	24.96M	24.96M	0.0	1.0	
C2 Pos Wid*	20.37ns	20.365714n	20.37n	20.37n	0.0	1.0	

Figure 4-10. RX_CLK and RX_D0 Timing in RGMII Align Mode

Note

Blue = RX_CLK, Purple = RX_D0



C2	1.0V/div	50Ω	BW:8.0G
C3	1.0V/div	50Ω	BW:8.0G

t1	0.0s
t2	4.0ns
Δt	4.0ns
1/Δt	250MHz

A' C3 ∫ 1.18V

20.0ns/div 6.25GS/s 160.0ps/pt
Stopped Single Seq
1 acqs RL:1.25k
Auto June 27, 2023 10:06:17

	Value	Mean	Min	Max	St Dev	Count	Info
C2 Freq	24.93MHz	24.930748M	24.93M	24.93M	0.0	1.0	
C2 Pos Wid*	20.28ns	20.276825n	20.28n	20.28n	0.0	1.0	

Figure 4-11. RX_CLK and RX_D0 Timing in RGMII RX Shift Mode

Note

Blue = RX_CLK, Purple = RX_D0

Table 4-9. RGMII Input Timing Specifications

PARAMETER	TEST CONDITION	Min	Typ	Max	Unit
T_{cyc}	TX_CLK / Clock Cycle Duration	36	40	44	ns
$T_{setup(align)}$	TX_D[3:0], TX_CTRL setup to TX_CLK (align mode)	1	2		ns
$T_{hold(align)}$	TX_D[3:0], TX_CTRL hold to TX_CLK (align mode)	1	2		ns

Table 4-10. RGMII Output Timing Specifications

PARAMETER	TEST CONDITION	Min	Typ	Max	Unit
$T_{skew(align)}$	RX_D[3:0], RX_CTRL delay from RX_CLK (align mode)	-750		750	ps
$T_{skew(shift)}$	RX_D[3:0], RX_CTRL delay from RX_CLK (shift mode enabled, default)	2			ns
T_{cyc}	RX_CLK / Clock Cycle Duration	36	40	44	ns
Duty_G	RX_CLK / Duty Cycle	45	50	55	%
T_r/T_f	RX_CLK / Rise, Fall Time (20% to 80% with C_{load}=5pF)			1.2	ns

Reduced Media Independent Interface (RMII)

Two separate configurations exist when using RMII mode: RMII master mode and RMII slave mode. In RMII master mode, the PHY is provided with a 25Mhz input clock on its XI pin and outputs a 50Mhz clock signal on its RX_D3 pin (pin 23) to send to the MAC. In RMII slave mode, the PHY receives a 50Mhz clock signal on the XI pin from either the MAC or external oscillator. It is imperative that the correct RMII mode is selected via bootstrap (this cannot be changed by register write) so the PHY expects the correct reference clock (either 25Mhz or 50Mhz). Refer to the DP83TC812 Schematic Checklist strap tool to ensure the correct RMII mode is selected. Next, probe the CLKOUT pin of the PHY and ensure a 25Mhz signal is seen if RMII master mode is used and a 50Mhz signal is seen if RMII slave mode is used.

The signals used for the RMII protocol are shown below.

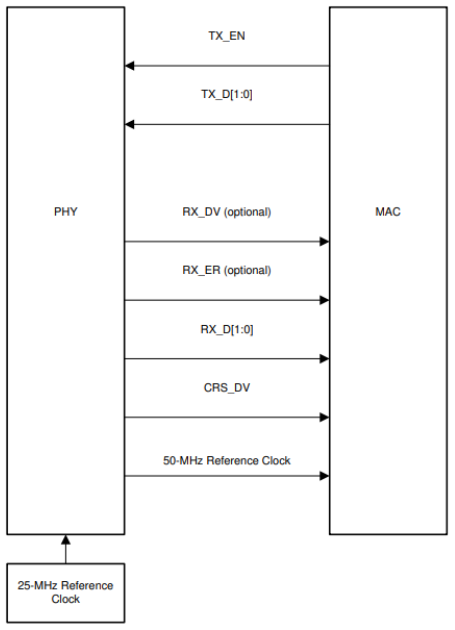


Figure 4-12. RMII Master Signaling

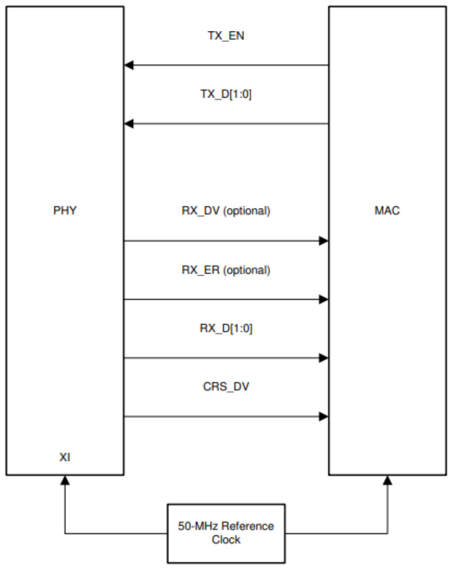


Figure 4-13. RMII Slave Signaling

4.12 Verify Open Alliance PMA Compliance

The IEEE802.3bw standard specifies several electrical tests to maintain proper transmitter electrical specifications of 100Base-T1 PHYs. These tests can be used to confirm the components and layout of a design are meeting requirements and not affecting the signal quality. See SNLA389 for more details on performing these tests. Note the script given in SNLA389 must be written at all times during regular operation of the PHY, not just for compliance testing.

For designs with link issues or CRC errors determined to be stemming from the MDI, performing these compliance tests can give additional insight on the issue. For example, a design failing the PSD test can point to the MDI traces being very long and having increased insertion loss. A design failing the jitter test can point to cross talk from nearby switching signals. Please refer to the DP83TC812 Schematic Checklist for recommended layout rules.

Most oscilloscope vendors have software and fixtures to run the 100Base-T1 PMA compliance tests. One such option can be found here. Below is a list of PMA compliance tests, and corresponding PHY test modes required.

Table 4-11. 100Base-T1 PMA Compliance Tests

PMA Compliance Test	Required PHY Test Mode
Transmitter Output Droop	Test Mode 1
Transmitter Clock Frequency	Test Mode 2
Transmitter Timing Jitter	Test Mode 2
Transmitter Distortion	Test Mode 4
Peak Differential Output	Test Mode 5
Transmitter Power Spectral Density (PSD)	Test Mode 5
MDI Return Loss	n/a
MDI Mode Conversion	n/a

4.13 Tools and References

4.13.1 DP83TC812 Register Access

If register access is not readily available in the application, USB-2-MDIO GUI is available from TI and can be used with an MSP430 Launchpad. The GUI supports reading and writing registers as well as running script files. The GUI can be used with the DP83TC812 and all other TI Ethernet PHYs. The USB-2-MDIO User's Guide and GUI are available for download here.

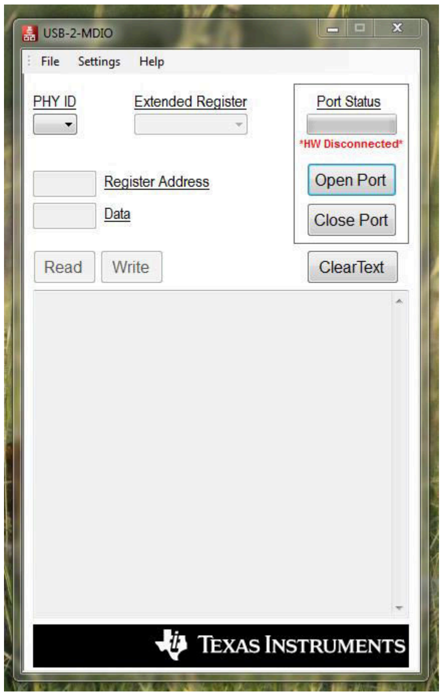


Figure 4-14. USB-2-MDIO GUI

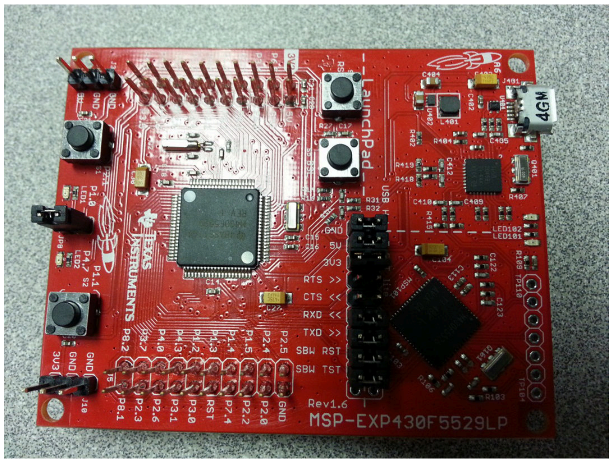


Figure 4-15. MSP430 LaunchPad

Below is an example script that can also be found in the USB-2-MDIO GUI in the Help menu:

```
// This is how you make a comment. All scripts must start with 'begin'
begin
// To read a register, all you need to do is put down the 4 digit
// HEX value of the registers (from 0000 to FFFF)
// Example to read registers 0001, 000A, and 0017
0001
000A
0017
// To write a register, all you need to do is put down the 4 digit
// HEX value of the register (from 0000 to FFFF) followed by the
// HEX you desire to configure the register to (from 0000 to FFFF)
// Example to write 2100 to register 0000 and
// Example to write 0110 to register 0016
0000 2100
0016 0110
// You must end the script by adding 'end' once you are finished
end
```

The Serial Management Interface defined by IEEE 802.3 is a single master bus. The MDC clock is generated by the bus master, typically an Ethernet MAC. To use the USB-2-MDIO GUI, connections must be made directly between the MSP430™ Launchpad and the DP83TC812 MDIO and MDC pins.

- Launchpad Pin 4.2 → PHY's MDIO Pin
- Launchpad Pin 4.1 → PHY's MDC Pin

4.13.2 DP83TC812 USB2MDIO Scripts

The following link contains a zip folder of USB-2-MDIO scripts that can be used to assist in the debugging process of the DP83TC812: DP83TC812 USB2MDIO Scripts

Such scripts include the SNLA389 given initialization scripts for master and slave, loopback scripts, test mode scripts for compliance testing, etc.

4.13.3 Extended Register Access

To read and write registers in extended register space, refer to the following procedures:

Write procedure for MMD "1F" registers:

write reg<000D> = 0x001F

write reg<000E> = <address>

write reg<000D> = 0x401F

write reg<000E> = <value>

Read procedure for MMD "1F" registers:

write reg<000D> = 0x001F

write reg<000E> = <address>

write reg<000D> = 0x401F

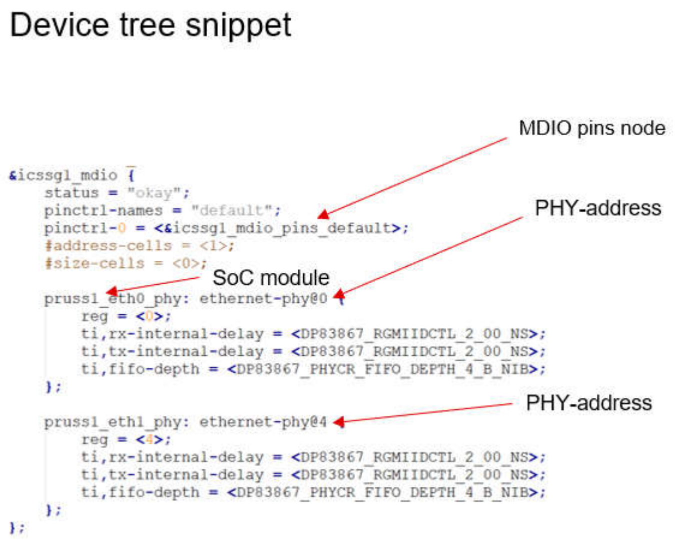
read reg<000E>

Note

To read/write MMD "1" registers, replace 1F with 01.

4.13.4 Software and Driver Debug on Linux

The two essential components required for the PHY to function on a Linux system are the device tree and driver file, for which the DP83TC812 drivers can be found here. Below is a sample format of what a device tree looks like along with labeling to clarify what each line means. The DP83867_PHYCR_FIFO_DEPTH_4_B_NIB and DP83867_RGMIIDCTL_2_00_NS variables are shared among all TI PHYs.



```
&icssg1_mdio {
    status = "okay";
    pinctrl-names = "default";
    pinctrl-0 = <&icssg1_mdio_pins_default>;
    #address-cells = <1>;
    #size-cells = <0>;

    pruss1_eth0_phy: ethernet-phy@0 {
        reg = <0>;
        ti,rx-internal-delay = <DP83867_RGMIIDCTL_2_00_NS>;
        ti,tx-internal-delay = <DP83867_RGMIIDCTL_2_00_NS>;
        ti,fifo-depth = <DP83867_PHYCR_FIFO_DEPTH_4_B_NIB>;
    };

    pruss1_eth1_phy: ethernet-phy@4
        reg = <4>;
        ti,rx-internal-delay = <DP83867_RGMIIDCTL_2_00_NS>;
        ti,tx-internal-delay = <DP83867_RGMIIDCTL_2_00_NS>;
        ti,fifo-depth = <DP83867_PHYCR_FIFO_DEPTH_4_B_NIB>;
    };
};
```

Figure 4-16. Ethernet PHY Device Tree Sample

4.13.4.1 Commonly Seen Linux Terminal Outputs

Using the terminal command "dmesg | grep mdio", there can be several clues on what's causing the PHY to not function appropriately from a software standpoint.

```
$ dmesg | grep "mdio"
```

One of the possible outputs is as follows:

```
$ mdio_bus xxx.ethernet-x: MDIO device at address 8 is missing
```

This message indicates that the PHY is not found on the MDIO bus, which can be caused by several issues, the most common one being a missing or incorrect device tree, but can also be due to a non-functional PHY or a bad SMI connection.

Once the PHY can be detected on the MDIO bus, another common error message is as follows:

```
$ Generic PHY xxx.ethernet-x: attached PHY driver [Generic PHY]
```

This message indicates that the driver file for the corresponding PHY is not loaded correctly or not present at all, and Linux loaded in a generic driver that most likely won't work with the PHY. In that case, verify that the driver successfully compiled and added to Linux, making sure the driver matches with the model of PHY used.

Finally, a message like this can display:

```
$ am65-cpsw-nuss c000000.ethernet eth3: PHY [c000f00.mdio:05] driver [TI DP83TC812CS2.0] (irq=POLL)
```

This message shows that the PHY has the correct driver loaded and is detected successfully. Run ifconfig to verify the network interface is present.

Example ifconfig output when the PHYs are successfully recognized as network adapters:

```
root@j7-evm:~# ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500  metric 1
        ether 24:76:25:a2:62:8b  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536  metric 1
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 82  bytes 6220 (6.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 82  bytes 6220 (6.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

5 Conclusion

This application note provides a suggested flow for debugging several common issues with the DP83TC812 such as lack of packet transfer and link issues. This guide provides a starting place but cannot encompass every such debug. Please reach out to your TI representative if the above does not resolve the issue.