Application Note

DP83TC812-Q1 TC10 System Timing Measurements

David Martinez, Elizabeth Higgins *Systems Engineering and Marketing*

ABSTRACT

This document describes the setup and measurements of TC10 protocol for a complete automotive Ethernet system that utilizes the DP83TC812-Q1 Ethernet Physical Layer (PHY).

Table of Contents

List of Figures

List of Tables

1 Introduction

In today's automotive industry, vehicles – especially electric vehicles (EVs) – are met with the problem of excessive battery consumption from electronic systems. This issue can be mitigated through partial networking using TC10 protocol. TC10 allows Ethernet nodes and their respective electronic control units (ECU) to be disabled when not in use which leads to longer drive range and reduced battery consumption.

The Open Alliance TC10 specification is an automotive Ethernet standard which defines a sleep mode and wake-up mechanism. To learn more about the general TC10 Sleep and Wake Sequence, see the *What is the TC10 automotive Ethernet standard and why is it important?* video.

This document demonstrates measurements for a complete Ethernet system that utilizes the DP83TC812-Q1, a 100BASE-T1-compliant Ethernet PHY with TC10 sleep and wake functionality. When a sleeping ECU receives a TC10 wake-up pulse (WUP), the ECU must wake and establish the Ethernet link in a reasonable amount of time. Thus, the sleeping ECU can be awakened and perform the requested function with minimal delay. This application note analyzes the steps a TC10 system goes through to wake up and link the Ethernet PHYs as well as detailing any possible optimizations for faster TC10 wake-up times. For more information on the DP83TC812-Q1 PHY configuration for TC10, see the *DP83TC812, DP83TC813: System Implementation of Open Alliance TC10 Sleep/Wake-up* application note.

1.1 Acronyms

This section defines the acronyms used in this document.

MDI	Medium Dependent Interface
MDIO	Management Data Input/Output
MDC	Management Data Clock
PHY	Physical Layer
PCS	Physical Coding Sublayer
PMA	Physical Medium Attachment
MAC	Medium Access Control

2 TC10 Test Setup

2.1 Overview

A test board was used to replicate a sleeping vehicle ECU which is referred to as Link Partner 2 (LP2). LP2 includes a LM62460-Q1 buck converter, LP8762-Q1 power management integrated circuit (PMIC), AM273x-Q1 Arm® based MCU, and the DP83TC812-Q1 TC10-compliant automotive Ethernet PHY. LP2 was used in conjunction with the DP83TC812EVM-MC to establish a link between the two devices, enter sleep mode, and then perform a remote TC10 wake sequence. The DP83TC812EVM-MC is referred as Link Partner 1 (LP1). Figure 2-1 shows how these components behave as a system.

2.2 Wakeup to Linking Sequence

The components and steps that precede the LP1 WAKE pin going high are particularly important because these components and steps decide how fast the system is going to wake up and establish an Ethernet link to begin communication. Overall, the hardware sets a strict limit on the wake-up time. Software also impacts this time, but the time software adds to the wake-up sequence varies with software optimization.

The following steps explain the wakeup to linking sequence:

1. LP1 WAKE pin goes HIGH, pulling LP1 INH pin HIGH.
2. LP1 transmits WUP to the LP2.
3. The LP2 PHY INH pin and the 1st stage enable of the buck (LM62460-Q1) rises.
4. Buck nRESET and the enable of the PMIC (LP8762-Q1) rises.
5. nRSTOUT of the PMIC and nRESET of the AM273x-Q1 rises. The MCU boot up starts.
6. The MCU management data input/output (MDIO) configures the PHY as master. The PHYs begin their link-up sequence.
7. LED0 on both boards lights up when both PHYs are linked.

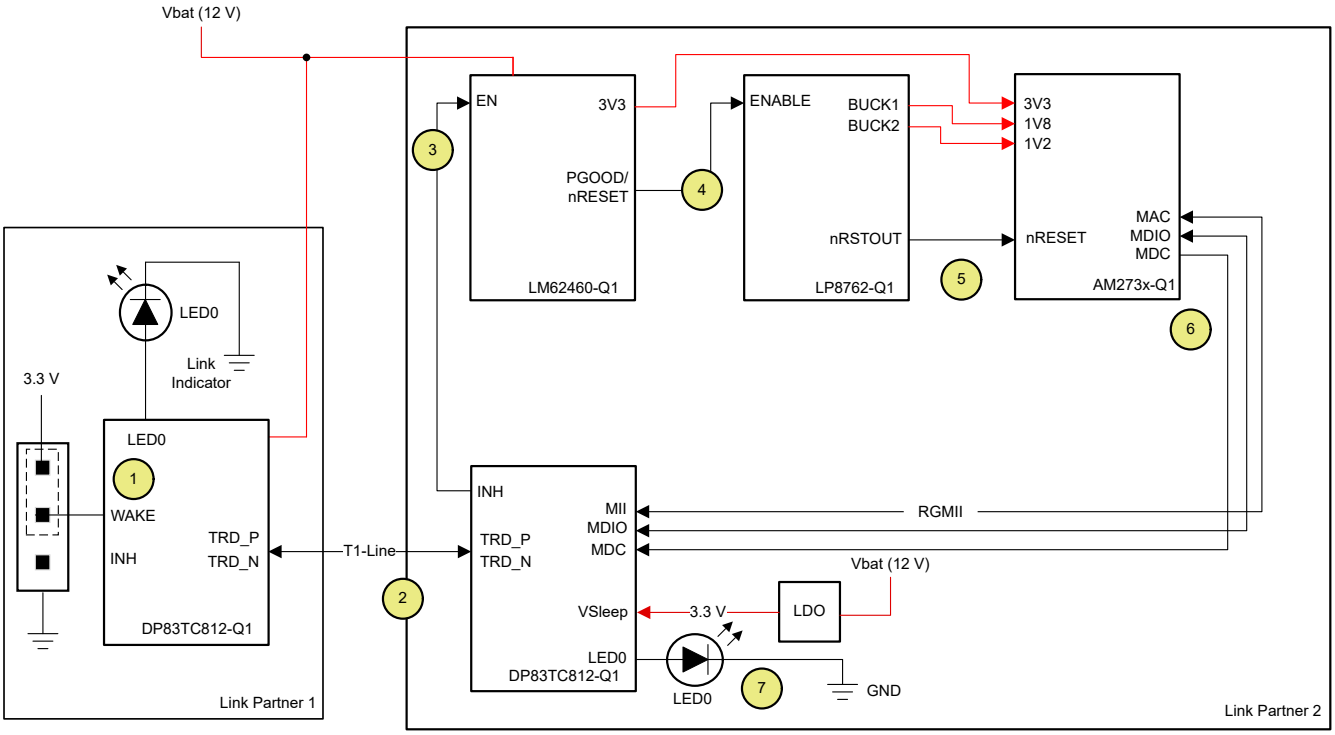


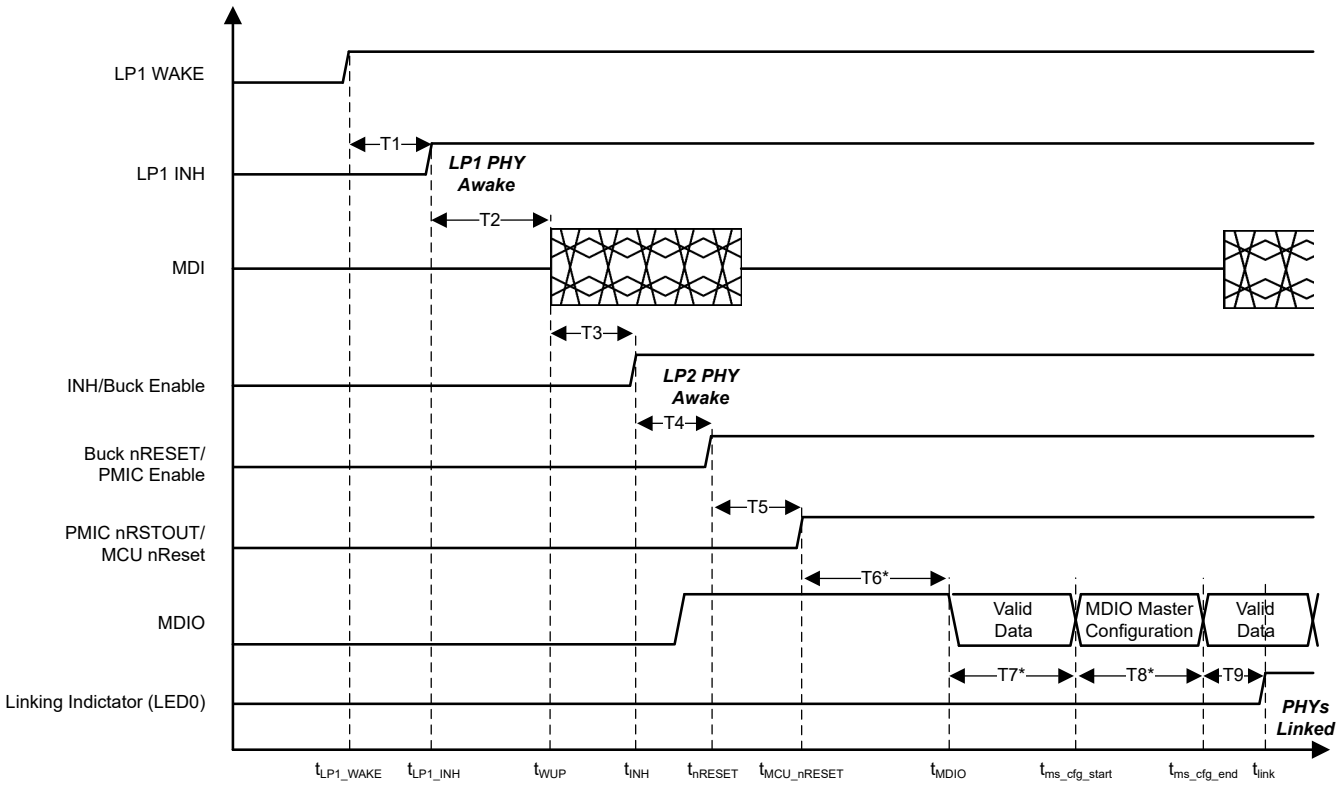
Figure 2-1. System Wake-up Sequence

For this test, both PHYs are bootstrapped as slave; however, the AM273x-Q1 configures the PHY of LP2 to be a master so the devices only link after the AM273x-Q1 is on and ready to communicate. This action is to prevent packet loss due to both PHYs being awake and linked without the MCU being ready to receive packets. Therefore, the linking process can occur only after the AM273x-Q1 has finished the boot-up sequence and has configured the PHY as a master through MDIO communication.

3 Measurement Summary

3.1 Complete Timing Diagram

Figure 3-1 illustrates the timeline of each measurement. The wake-up sequence is broken down in steps from T1 to T9. T6, T7, and T8 are marked with "*" (an asterisk) because they are dependent on software. Therefore, T6, T7, and T8 are areas where any design can look to further optimize the TC10 timing. The other time intervals are based only on the hardware of the system.



MDIO rises initially because MDIO is tied to the LM62460-Q1 3V3 output.

Figure 3-1. Wake to Link Timing Diagram

3.2 Measurement Summary

Table 3-1 details the measurements.

Table 3-1. Summary of Measurements

Interval	Time Marker	Description	Measurements
	$t_{link} - t_{LP1\_WAKE}$	LP1 WAKE going high to linking (System wake to link time)	33.32ms + T*
T_1	$t_{LP1\_INH} - t_{LP1\_WAKE}$	LP1 WAKE going high to LP1 INH going high (LP1 wake-up sequence started)	20.6µs
T_2	$t_{WUP} - t_{LP1\_INH}$	LP1 INH going high to WUP being transmitted	10.53ms
T_3	$t_{INH} - t_{WUP}$	WUP being transmitted to LP2 INH going high (LP2 wake-up sequence started)	45.8µs
T_4	$t_{nRESET} - t_{INH}$	LP2 INH going high to buck nRESET going high (Buck ready)	6.00ms
T_5	$t_{MCU\_nRESET} - t_{nRESET}$	Buck nRESET going high to MCU nRESET going high (PMIC ready)	2.184ms
T_6^*	$t_{MDIO} - t_{MCU\_nRESET}$	MCU nReset going high to MDIO communication starting (MCU start-up)	*
T_7^*	$t_{ms\_cfg\_start} - t_{MDIO}$	MDIO communication starting to master configuration start	*
T_8^*	$t_{ms\_cfg\_end} - t_{ms\_cfg\_start}$	Master configuration time	3.914ms*
T_9	$t_{link} - t_{ms\_cfg\_end}$	End of master configuration to linking	14.541ms

3.3 LP1 Wake to Linking Time

The time from LP1 WAKE rising to linking can be estimated from the following function:

$$t_{link} - t_{LP1\_WAKE} = (T_1 + T_2 + T_3 + T_4 + T_5 + T_9) + (T_6^* + T_7^* + T_8^*) \quad (1)$$

$$t_{link} - t_{LP1\_WAKE} = 33.32ms + (T_6^* + T_7^* + T_8^*) \quad (2)$$

$$T^* = (T_6^* + T_7^* + T_8^*) \quad (3)$$

$$t_{link} - t_{LP1\_WAKE} = 33.32ms + T^* \quad (4)$$

Note

T* is a variable in TI's TC10 timing measurement to summarize the times that can be optimized with software. These recommendations are discussed in Section 5.1.3.

4 Timing Measurements

4.1 LP1 WAKE to INH (T_1)

Figure 4-1 illustrates that when the WAKE pulse of the LP1 PHY goes HIGH, the INH pin is driven HIGH in 20.60µs.

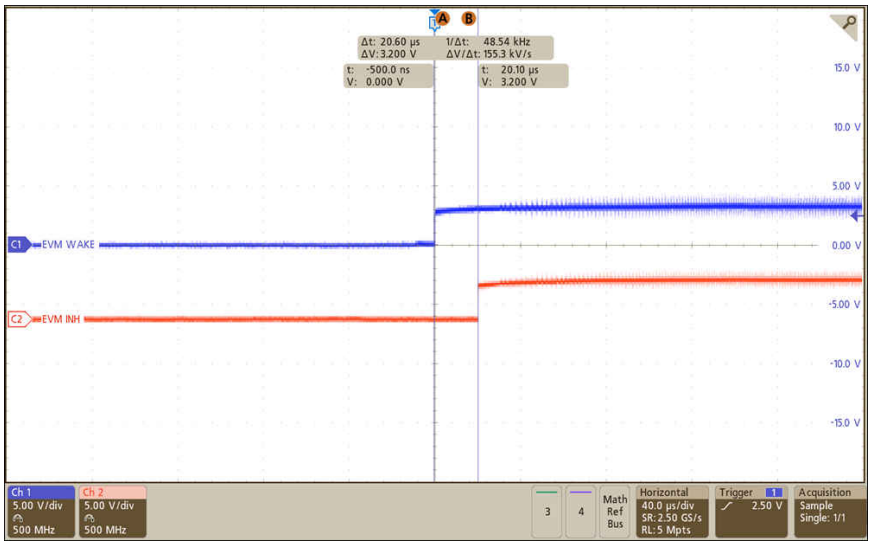


Figure 4-1. LP1 WAKE to INH

The DP83TC812EVM-MC was modified by removing the INH pin connection to the MCU GPIO pin. This removal prevents a slower INH rise time by removing the additional leakage path of the undefined MCU GPIO pin. Alternatively, a resistor can be added between the INH and MCU GPIO pin to limit the leakage current.

4.2 LP1 INH to WUP (T_2)

Figure 4-2 shows that when the INH pin is HIGH, the LP1 PHY is awake and sends a WUP to the LP2 PHY in 10.53ms.

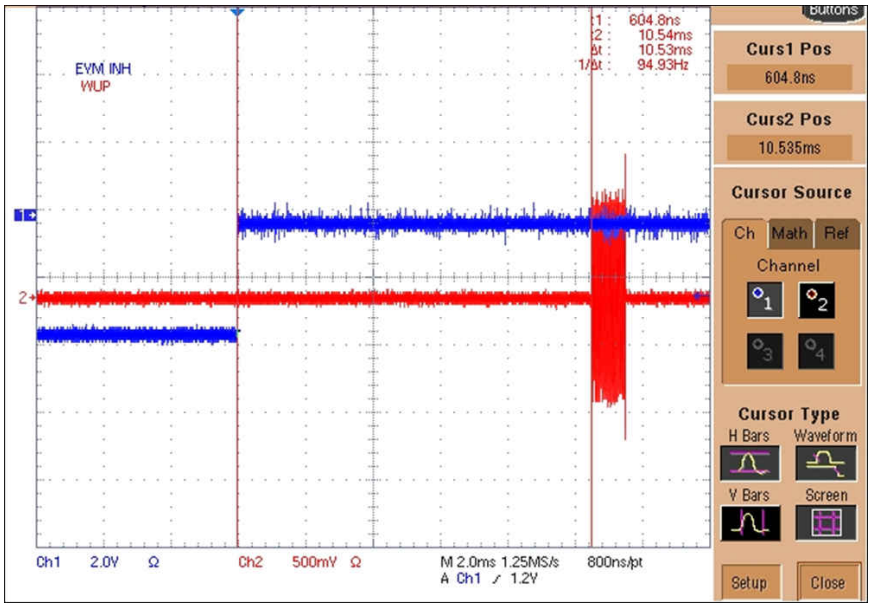


Figure 4-2. LP1 INH to WUP

4.3 WUP to PHY INH (T_3)

Figure 4-3 shows that upon receiving the WUP on the T1-line, the LP2 PHY detects the energy and pulls the INH HIGH. The time of interval T_3 is 45.8µs.

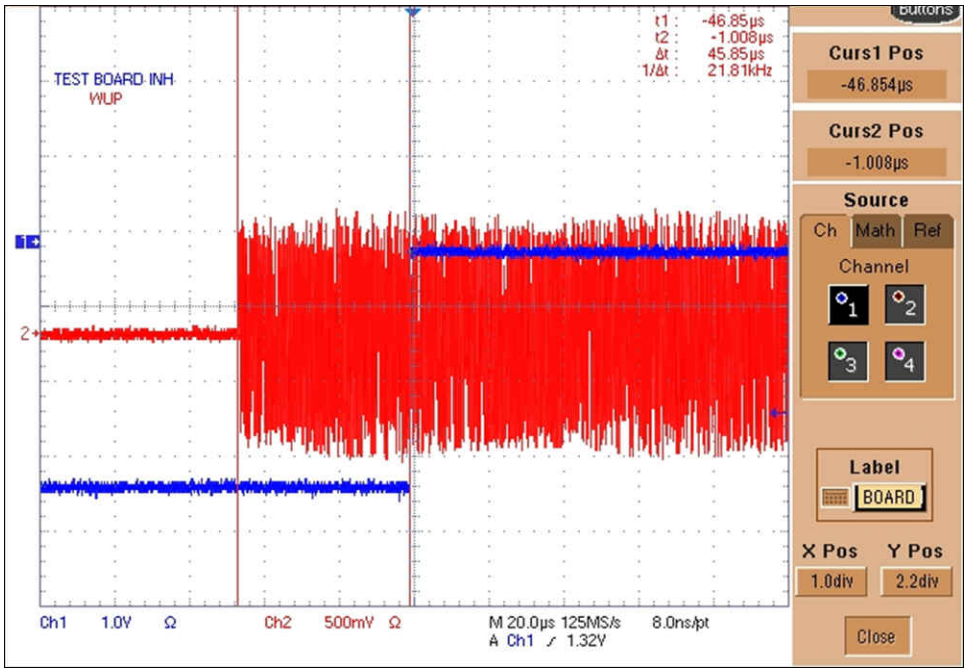


Figure 4-3. WUP to INH

4.4 PHY INH/Buck EN to Buck nRESET (T_4)

The time it takes for the enable pin of the 1st stage buck to drive the nRESET pin HIGH was measured to be 6ms (see Figure 4-4).

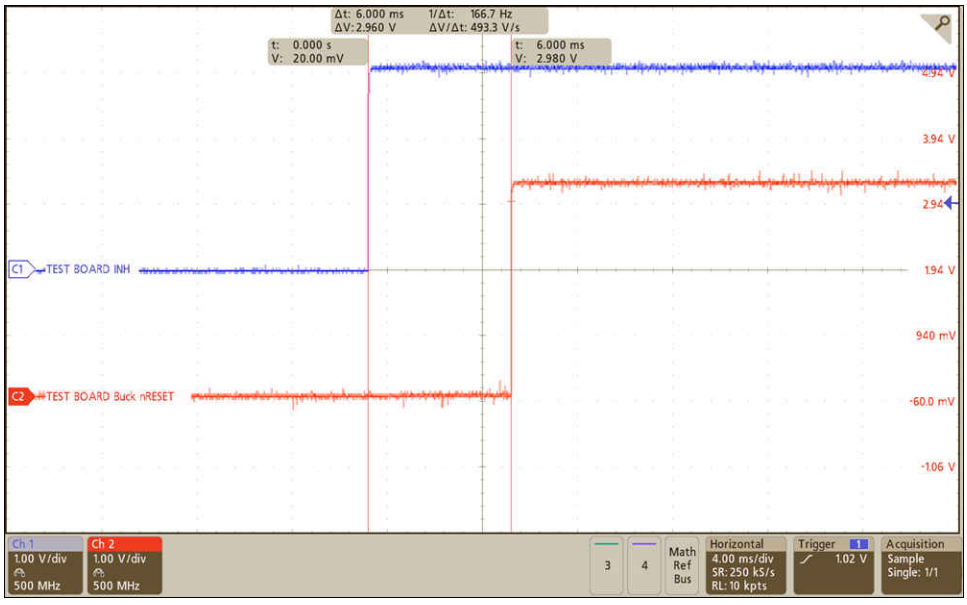


Figure 4-4. Buck EN to nRESET Signal, or to PMIC Enable Signal

4.5 Buck nRESET/PMIC Enable to MCU nReset (T_5)

Both the PMIC enable and MCU nReset are tied to the GPIO pins of the PMIC. Figure 4-5 shows the interval for these two waveforms. The time of interval T_5 was measured to be 2.184ms.

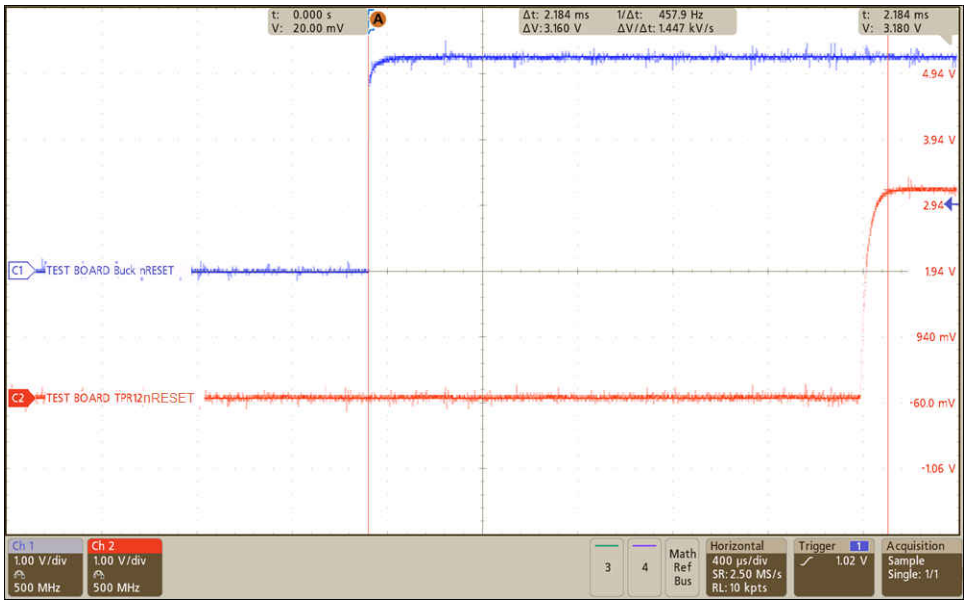


Figure 4-5. Buck nRESET/PMIC Enable to MCU(TPR12) nReset

4.6 MCU nReset to MDIO Communication (T_6 and T_7)

The AM273x-Q1 starts the boot-up process once the nReset goes HIGH. When booted, the MDIO eventually begins the master configuration depending on when the configuration is started in software. As Section 5.1 shows, these steps are dependent on the software. Therefore, for times T_6 and T_7, it is more beneficial to view recommendations for improving TC10 wake-up time.

4.7 MDIO Master Configuration + Linking (T_8 and T_9)

Configuring the PHY as the master involves writing to a series of registers as a minimum requirement to be compliant with Open Alliance Specifications. For more information about these specifications, see the *DP83TC812, DP83TC813, and DP83TC814: Configuring for Open Alliance Specification Compliance* application note.

Table 4-1 is a timeline for the data written along the MDIO line. Overall, decoding the data transmitted shows that interval T_8 – the time it takes to configure the PHY as master – is 3.914ms for this system.

After the PHY is successfully configured as master, both master and slave PHYs begin a handshake process to establish a link. The echo canceler, scrambler, equalizer, and timing of the PHY undergo training to provide proper communication. Interval T_9 is measured to be 14.541ms.

Table 4-1. MDIO Timeline for Master Configuration + Linking

Time (ms)	Register	Data (Write)	Action
0			$t_{ms\_cfg\_start}$: MDIO Master Configuration Starts
0.010	0x001F	0x8000	Hard Reset
0.225	0x0523	0x0001	Disable Linkup
0.396	0x0834	0xC001	Configure PHY as Master
0.396–2.370			Configurations to enable shorter link-up time
2.370	0x001F	0x4000	Soft Reset
2.584	0x0523	0x0000	Enable Linkup
3.914	0x001F	0x4000	$t_{ms\_cfg\_end}$: Soft Reset
18.455			t_{link}: LED0 lights up. PHYs are linked.

5 Measurement Evaluation

5.1 Recommendations for Optimizing Variable TC10 Times

The following section specifies how T_6^* , T_7^* , and T_8^* can be optimized to decrease the time from PHY wakeup to successful linking.

5.1.1 Improving MCU Boot-up Time (T6)

A critical time to minimize is the MCU boot-up time specified for a system. This time is important because the PHY master configuration does not start until the MCU is booted.

In addition to optimizing the software used for an MCU as thoroughly as possible, use the fast boot option of the MCU, if the feature is available. For example, the AM273x-Q1 can decrease the boot-up time by:

1. Minimizing a configurable reset delay
2. Minimizing the configurable delay for boot up after nRESET goes HIGH. If the device is operating from an oscillator source instead of a crystal source, faster start-up is possible.

5.1.2 Improving MDIO State Machine (T7)

The interval $t_{link} - t_{MDIO}$ is dependent on when the PHY master configuration actually begins. Optimizing T7 allows T8 and T9 to occur sooner. Keeping T7 idling for too long only increases the PHY link time.

Reducing T7 depends on the implementation of the PHY state machine in the MCU. The PHY state machine defines the steps before and after the master configuration when the MDIO is active. As an example, Figure 5-1 shows the MDIO state machine for the AM273x-Q1 SDK.

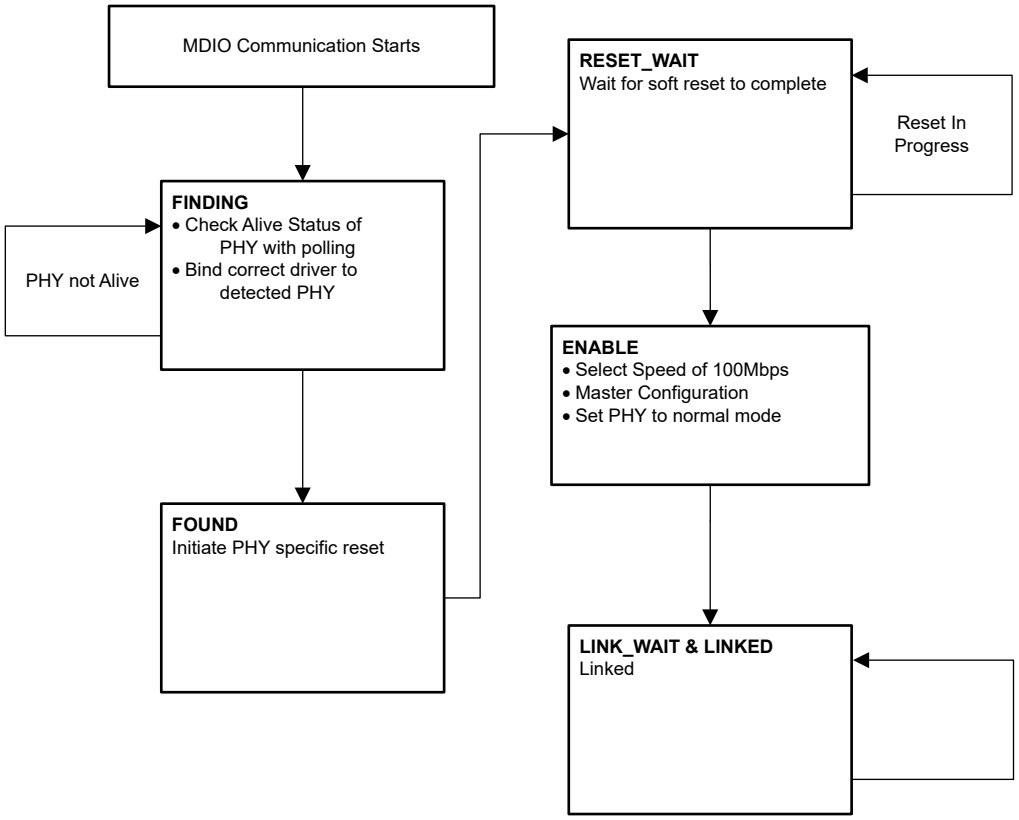


Figure 5-1. MDIO State Machine for AM273x-Q1 SDK

Possible optimizations of this state machine include:

- The AM273x-Q1 software development kit (SDK) contains many drivers for various PHYs. However, the AM273x-Q1 can be optimized to use the DP83TC812-Q1 driver to save time.
- The DP83TC812-Q1 always transmits or receives data at a rate of 100Mbps. Selecting a speed of 100Mbps is not necessary.

5.1.3 Optimizing MDIO Timeline (T8)

Table 4-1 shows the timeline for the master configuration. Note that if a MAC interface communicates with only one PHY, a feature like polling is not necessary because the MAC always communicates with the same PHY address. This feature delays the master configuration as the MAC has to constantly read the alive status of each PHY address.

5.1.3.1 Optimizing Master Configuration by Removing Polling

Each MDIO frame is 64 bits since the MDIO data follows Clause 22. The data format for clause 22 is in the following list.

- 32-bit preamble (optional)
- 2-bit start
- 2-bit opcode (Write and Read operations only)
- 5-bit PHY address
- 5-bit register address
- 2-bit turnaround time
- 16-bit data

Overall, the master configuration sequence in Table 4-1 takes 179 read or write instructions with polling. The MDC was measured to be 3 MHz.

By removing polling, the master configuration time is decreased since polling creates unnecessary read instructions. When removing all unnecessary read instructions, only 90 instructions are needed to configure the PHY as master. Most of these write instructions involve extended registers, which require 4 writes to successfully write to them.

Removing the optional preamble is also beneficial because the preamble is not needed for the read and write transactions. This decreases the size of each MDIO frame to 32 bits.

Δt_{ms_cfg}= 90 instructions × 32 bits × (1/3 MHz) = 0.960ms. This is 2.954ms less than the actual time measured.

5.1.3.2 Optimizing Master Configuration by Improving MDC

The DP83TC812-Q1 supports an MDC up to 25 MHz. If a MAC interface supports an MDC of 25 MHz, this time can be further optimized as the following equation shows.

Δt_{ms_cfg} = 90 instructions × 32 bits × (1/25 MHz) = 0.23ms. This is 3.799ms less than the actual time measured.

5.1.4 PHY Configuration During Sleep

Currently, the DP83TC812-Q1 always performs a hard reset after waking up from TC10 sleep so all the registers return to their default state.

The next generation 100BASET1 TI PHY features configuration of up to 20 registers immediately upon waking from sleep mode. This means that interval T7 and T8 are ignored entirely and the PHY can initiate link up with a link partner without waiting for configuration by the MCU. The sequence of events is shown in the following list:

1. Board MCU programs a list of register address + content in the PHY before the PHY goes to sleep mode
2. PHY goes to TC10 sleep mode
3. PHY receives WUP, driving INH HIGH and beginning the wake-up sequence as detailed in Section 2.2
4. PHY automatically configures the stored register addresses + content from retained memory
5. PHY initiates link up with a link partner

5.1.5 Other Configurable Values

The LP8762-Q1 PMIC has a configurable reset delay. Shortening this delay helps the MCU nRESET rise sooner, shortening the overall time from wake up to linking.

5.2 Alternative TC10 Test

Another TC10 test keeps the LP2 PHY bootstrapped to master mode. Specifically, LP2 immediately wakes up as master and LP1 wakes up as a slave.

To prevent link up before the MAC interfaces are ready to communicate, the PHYs can be bootstrapped in managed operation. In managed operation, commands through MDIO allow the PHY to exit standby mode and enable both the PCS and PMA so the devices can successfully link. Using this test, T7 and T8 are also ignored entirely, allowing for link training to begin from configuring a single register by MDIO.

6 Conclusion

TC10 is an Open Alliance standard that defines the wake-up and sleeping protocol for Ethernet PHYs. Integrating TC10 into an actual system is critical for power management systems due to the potential TC10 has to give vehicles options to reduce power consumption.

The idea of longer battery life only becomes more important as vehicles, especially EVs, push their power management systems to the limit.

7 References

1. Texas Instruments, *DP83TC812, DP83TC813, and DP83TC814: Configuring for Open Alliance Specification Compliance* application note
2. Texas Instruments, *DP83TC812, DP83TC813: System Implementation of Open Alliance TC10 Sleep/Wake-up* application note
3. Texas Instruments, DP83TC812EVM-MC: DP83TC812 evaluation module for 100BASE-T1 to 100BASE-TX media converter product folder
4. Texas Instruments, *What is the TC10 automotive Ethernet standard and why is it important?*, E2E™ forum

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265