

# AN11053

基于 NXP ARM 微控制器的以太网吞吐性能

Rev. 1-23/3/2011 应用手册

## 文档信息

信息	内容
关键字	LPC1700, 以太网, ARM7, ARM9, Cortex-M3
摘要	本文档讨论了在 LPC1700 上测量以太网吞吐性能的 3 种不同方案, 以及系统优化的详细情况。

深圳市伟博创科技有限公司

Tel:(0755)87240703 Fax:83240724

<http://www.weboch.com.cn>

<http://www.longdouble.com>

## 1. 简介

以太网是世界上应用最广泛的局域网技术。以太网从1980年代起就开始应用,已有IEEE 802.3 标准,规定了一系列传输速率。在嵌入式系统中,应用最普遍的是10Mbit/s 和100Mbit/s (就是通常所说的10/100 以太网)。

现在NXP(恩智浦半导体)有超过20款内置以太网功能的ARM MCU,覆盖了3代ARM架构(ARM7, ARM9, Cortex-M3)。NXP在3种架构中采用了相同的实现方法,所以在系统更新换代时,设计人员可以节省大量的时间和资源,

### 1.1 优异的实现方法

NXP的以太网部件(见图1)包含一个全功能10/100 MAC(媒体访问控制器),使用DMA硬件加速以提高性能。MAC完全符合IEEE 802.3标准,并使用MII(媒体独立接口)或RMII(精简MII)协议,通过片上MIIM(媒体独立接口管理)总线,与片外以太网PHY(physical layer 物理层)连接。

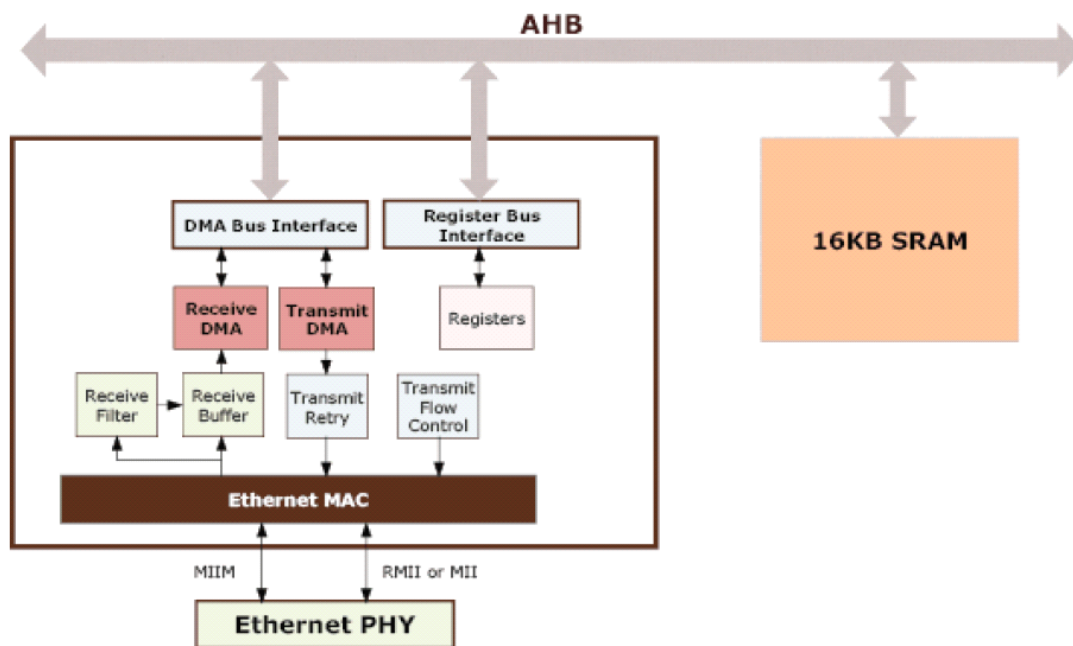


图1 LPC24xx 以太网部件

NXP以太网的优异性能如下所述:

#### (1) 全以太网功能

该部件支持以太网的所有操作,符合802.3标准。

#### (2) 增强型架构

NXP提高了架构的性能,增加了几个额外的功能,包括接收过滤、自动抵触返回(collision back-off)、帧重发、电源管理(通过部件选择)等等。

#### (3) DMA硬件加速

该部件有两个DMA管理器,分别负责发送和接收。使用分散收集DMA(Scatter-Gather DMA)

自动帧重发和接收，进一步卸载 CPU（offloads the CPU）。

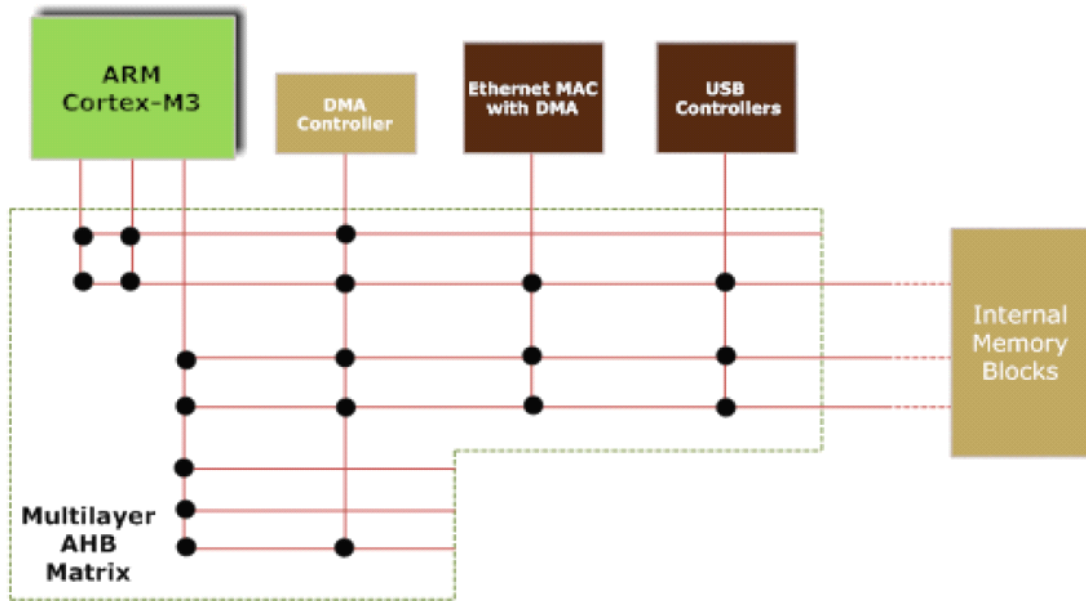


图 2 NXP Cortex-M3 架构

## 2. NXP LPC1700 微控制器的以太网吞吐性能

在以太网中，两个或更多站使用以太网协议通过公用通道发送或接收数据。对每一个网络要素（通道或站），以太网性能有不同的含义。带宽、吞吐量、延迟是衡量整体性能的重要指标。对于通道，用带宽来衡量连接性能，而吞吐量则代表可用数据在通道中发送的速率。对于站，以太网性能意味着设备在以太网通道中全速操作的能力。另一方面，延迟用于衡量几种因素（传送时间、处理时间、故障、重试等等）引起的迟滞时间。

本文档的着重点放在 NXP LPC1700 操作全速以太网通道的性能（通过以太网接口，由内部 EMAC 模块和外部 PHY 芯片提供）。在这种方式下，吞吐量定义为 MCU 每秒发送到通道或从通道接收的可用数据。这个概念适用于其他支持以太网的 NXP LPC 微控制器。不幸的是，这些测试通常都需要特定的设备，如网络分析器和/或网络通信发生器，以获得精确的测量结果。不过，使用简单的测试步骤获得估计值是可以做到的。实际上，我们的目的是搞明白影响以太网吞吐性能的不同因素，所以用户可以关注不同的技术，以改进以太网的性能。

这里，我们只考虑发送器的吞吐性能；而接收器会有一点复杂，因为它涉及到发送器发送信息到通道的性能（这种情况下，接收器的吞吐量受到发送器通道发送信息的吞吐量的影响）。我们只要得到发送器的吞吐数据，我们就能把这个数据当作接收器可以达到的最大理想数据（在理想条件下），就能获得与接收器的吞吐数据。

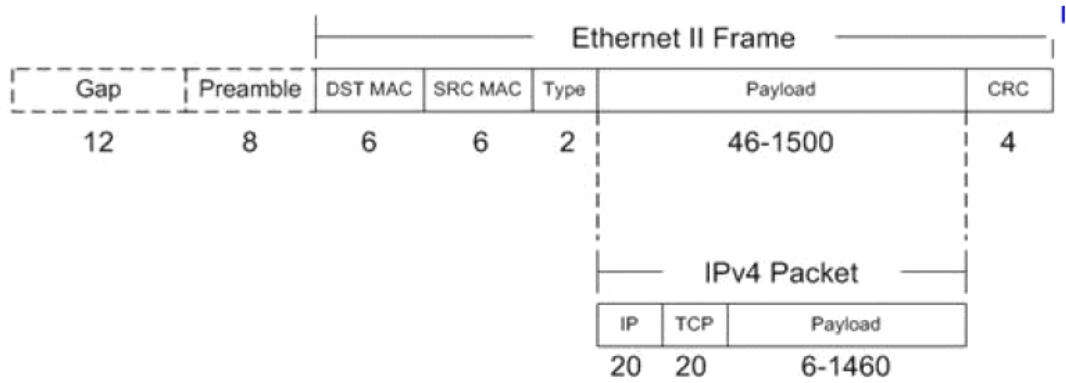


图3 以太网 II 格式

考虑 100Mbit/s 1 位的速率，每一帧 (frame) 包含负载 (可用数据，最小 46B，最大 1500B)，以太网的头部 (14B)，CRC (4B)，序文 (Preamble, 8B)，内部包缺口 (inter-Packet Gap, 12B)，以下为每秒最大可能帧和吞吐量：

最小帧：(46B 的数据) ->1448809 帧/秒->6.84MB/s

最大帧：(1500B 的数据) ->8127 帧/秒->12.19MB/s

以上的速率是实际中可能达到的最大可能值。这些值是理想情况下的值，实际中会比这小。

注意：(1) 帧/s 是通过以太网连接速度 (100M bit/s) 除以总的帧位大小计算得到的 (最小帧：84\*8=672，最大帧：1538\*8=12304)。

(2) MB/s 是通过帧/s 乘以每一帧可用数据的字节数得到的 (最小帧：46B，最大帧：1500B)。

## 2. 1 测试条件

MCU: LPC1768, 100MHz

Board: Keil MCB 1700

PHY chip: National DP83848 (RMII 接口)

工具链: Keil uVision4 v4.1

从 RAM 运行代码

TxDescriptorNumber=3

以太网模式: 全双工-100Mbit/s

## 2. 2 测试描述

为了得到最大吞吐量，将发送由 1514B (包含以太网头部)、75B 负载 (可用数据) 组成的 50 个帧。EMAC 控制器 (以太网控制器) 将自动添加上 CRC (4B)。为了测量这个处理过程所花费的时间，一个 GPIO (本例为 P0.0) 在发送帧时置位，发送完毕后立即清零。可以用示波器测量 P0.0 引脚上产生脉冲的宽度，这样就能得到相应的时间。板子通过以太网电缆连接到 PC 上。

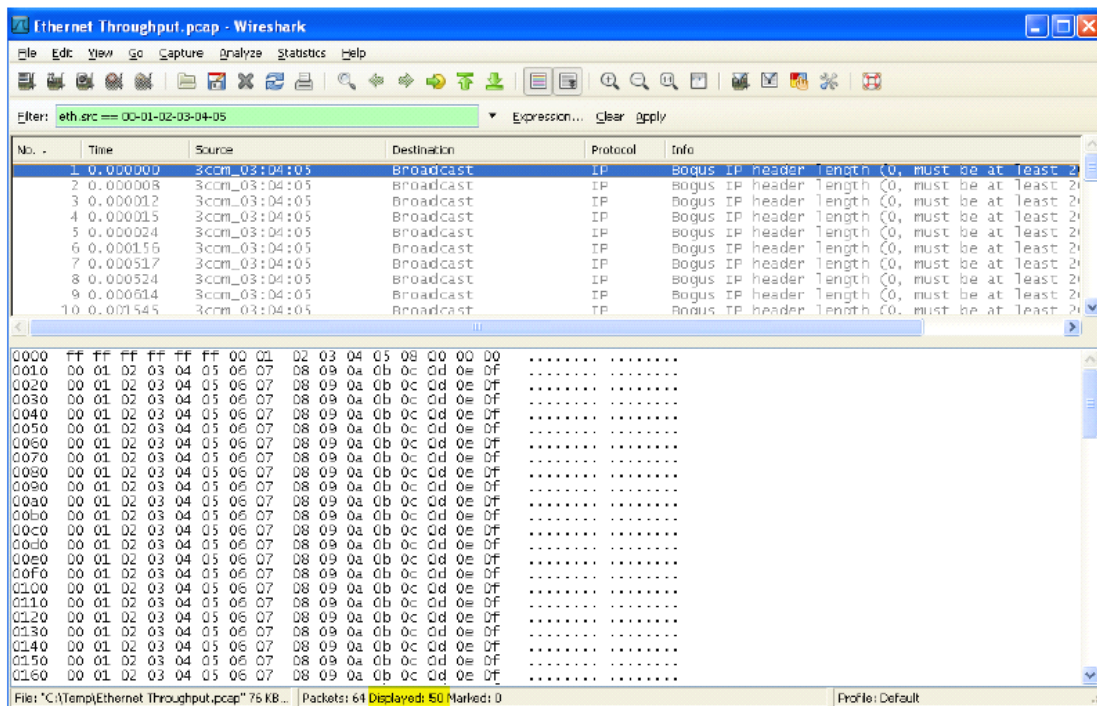


图 4 用 WireShark 检查数据

PC上运行一个嗅探器程序（本例中为WireShark <http://www.wireshark.org/>）来检查50个帧是否发送、数据是否正确。负载中有一个特殊的部分，一旦发生错误，就可以识别。50个帧到达PC时无错，则测试有效。

### 2. 3 测试

EMAC使用系列符号，提供了相应的指针以指示数据缓冲、控制和状态信息的存储位置。传送时，帧数据在进入数据缓冲器时被应用代替。EMAC使用DMA获得用户数据，并在发出之前装载帧负载。

如上所述，应用程序所使用的以复制数据到数据缓冲器的方法，将会影响吞吐量的测量。有三种情况：

- (1) 理想情况下，完全不考虑应用程序。
- (2) 典型情况，应用程序使用处理器复制数据到EMAC数据缓冲器。
- (3) 优化情况下，应用程序通过DMA复制数据到EMAC数据缓冲器。

#### 2. 3. 1 情况描述

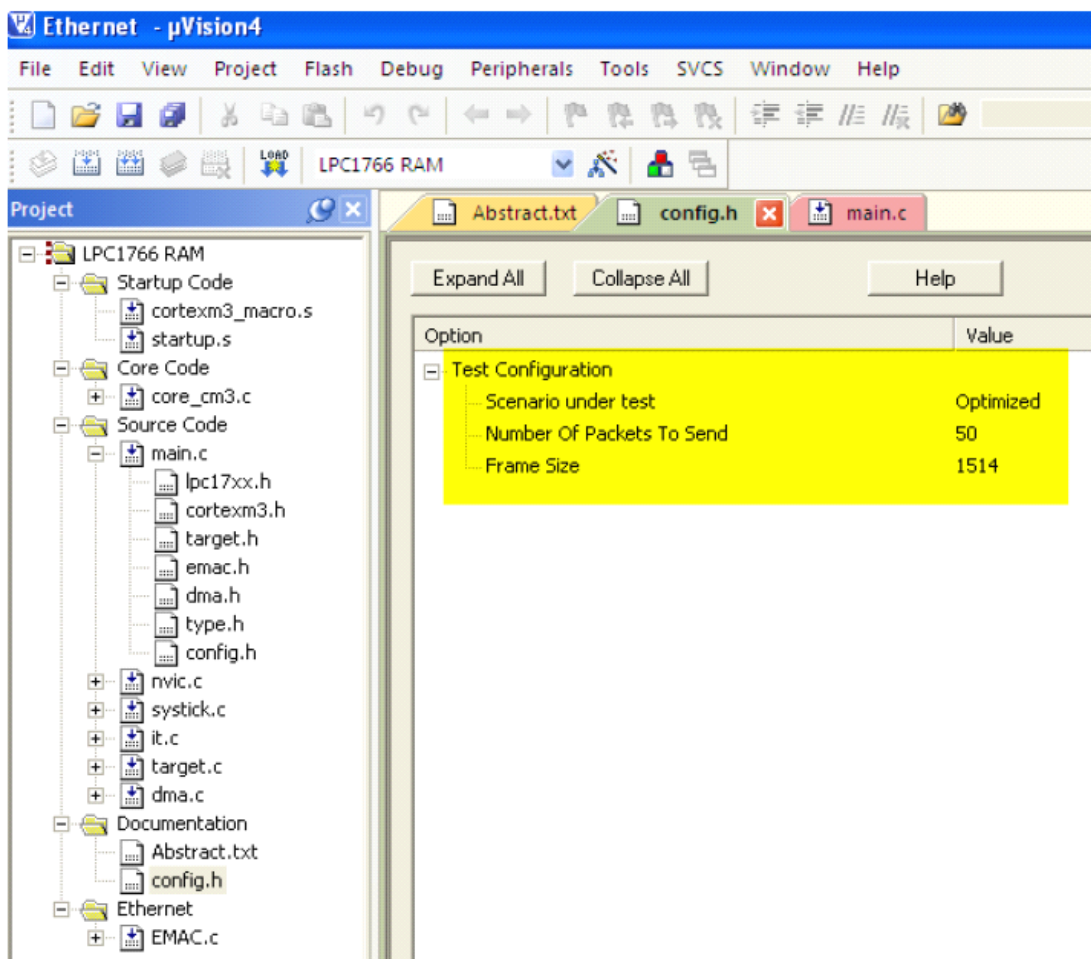
1. 理想情况：在这种情况下，软件用测试部件建立符号数据缓冲器，且只有TxProduceIndex增长50倍，触发帧的发送。换句话说，根本无需考虑应用，即使不是典型的用户情况，仍然可以在发送时提供最大的可能吞吐。

2. 典型情况：这种情况比较典型，在发送帧之前应用程序复制数据到符号数据缓冲器。与之前的结果相比，可以明显看到应用程序影响到整体性能。此种情况，不宜作为实际EMAC的吞吐量。然而，这个例子说明了未优化的应用程序如何降低整体性能，给人的印象就是硬件太慢了。

3. 优化情况：使用DMA，复制应用数据到符号数据缓冲器。这种情况下，采用优化的方法，体现出快速LPC1700的优点。

2.4 软件

本文档中，提供了Keil MDK工程形式的测试软件方案。



打开config.h文件，使用配置向导，可以选择想要的模式。除此之外，发送包的数量和帧的大小也可以在此文件中修改。

2.5 测试结果

测试结果做成表格如下所示：

Table 1. Throughput results

	Frames sent	Payload (Bytes)	Total data (Bytes)	Time (msec)	Throughput (MB/s)	% relative to max possible
<b>Max Possible</b>					<b>12.19</b>	<b>100%</b>
<b>Scenario 1</b>	50	1500	75000	6.25	12.00	<b>98.44%</b>
<b>Scenario 2</b>	50	1500	75000	10.44	7.18	<b>58.93%</b>
<b>Scenario 3</b>	50	1500	75000	7.1	10.56	<b>86.66%</b>

### 3. 结论

尽管情况 1 不符合实际，但它提供的最大可能值可以作为参考，它非常接近以太网 100Mbit/s 的最大可能值。情况 2，很明显影响到整体性能。最后，情况 3，通过优化可以改进整体的吞吐量。

从 FLASH 运行代码（而不是从 arm）也可以进行优化，且能得到更好的结果。

总之，以太网的吞吐量主要受传输数据到符号数据缓冲器的方式的影响。改进此过程，可以大大提高以太网的整体性能。LPC1700 以及其他 LPC 系统的芯片都可以进行优化（基于 DMA 的支持、EMAC 硬件和智能存储总线架构）。