

WBMS 统计命令解析及 PDR PS 测试方法

Yue, Tang

FAE/China CM

ABSTRACT

本文介绍了 TI WBMS 的统计功能，并详细说明了如何利用统计功能来测量 WBMS 系统型指标 PDR（Packet Delivery Rate）和 PS（Path Stability）。PDR，PS 指标比传统的 PER 能更好地衡量 WBMS 系统的稳定性。

Contents

1	引言	2
2	PDR 和 PS 的定义	2
	2.1 PDR（Packet Delivery Rate）	2
	2.2 PS（Path Stability）	2
3	TI WBMS SDK 统计命令集	2
	3.1 统计命令及其功能 简介	3
	3.2 基于统计命令的 PDR/PS 计算	4
4	使用 LLCOM 工具测试 WBMS 系统的 PDR、PS	5
	4.1 LLCOM 工具介绍	5
	4.2 测试基本思路及详细步骤	6
	4.3 LLCOM PDR PS 测试的总结	12
5	参考文献	12

Figures

Figure 1.	LLCOM 主界面	5
Figure 2.	LLCOM 脚本页面	6
Figure 3.	TI CSU EVM 框图	6
Figure 4.	测试环境搭建	7
Figure 5.	WM 工程中开启 MANUAL_NETWORK_START	8
Figure 6.	选择串口及波特率	8
Figure 7.	快捷发送页面组网命令序列	9
Figure 8.	组网命令序列及其响应	10
Figure 9.	选择测试脚本	10
Figure 10.	逻辑分析仪测量通信间隔小于 100ms	11

1 引言

PER (Packet Error Rate) 是一个重要的射频指标，常用于评估射频芯片的接收灵敏度，但是用于评估 WBMS 整体系统性能时则不够全面，WBMS 系统除了射频芯片，还包含 AFE, HOST MCU 等，实际应用中客户更关心是整体系统是否能在 100ms (FTTI- Fault Tolerant Time Interval) 成功读取到电芯电压和温度，重传率是否足够低，因此 PDR (Packet Delivery Rate) 和 PS (Path Stability) 常被用于衡量 WBMS 系统的稳定性。

以下是本文提到的一些英文缩写的含义：

2 PDR 和 PS 的定义

2.1 PDR (Packet Delivery Rate)

PDR 指的是在 100ms 间隔内读取到电芯数据时的成功率，要求是不低于 99.9%。在 TI WBMS 协议栈中，例如一个 1 WM 对 15WD 的网络，每次电芯数据的读取是由 WM 主节点发起，15 个 WD 依次分时返回各自的数据，如果 15 个 WD 有一个或多个 WD 在这次通讯中返回数据失败，则 WM 会自动重传，直到所有 15 个 WD 均成功返回数据，重传最大次数默认为 5 次，如果超过最大重传次数仍不成功，视为一次通讯失败。考虑到 100ms 间隔要求，并且对于 15WD 的网络，每次通讯需要花费 21ms 左右，因此最大重传次数不能大于 3。实际测试中通常会进行 10000 次电芯采样通讯来衡量这一指标。PDR 也包括系统 PDR 和各个无线节点的 PDR, 系统 PDR 如能达到 99.9%，节点 PDR 则一定能达到。

2.2 PS (Path Stability)

PS 强调的是点对点通讯成功率，需要指出的是每次重传也被记为一次通讯。例如一个 1 对 3 的网络，第一次 WM 发送电芯采样时，3 个 WD 均成功反馈，第二次采样时，WD 2 没有成功反馈，因此触发重传，而在这次重传中所有 3 个 WD 数据传递均成功。这个例子中，虽然只进行了两次数据采集，但实际发生了三次通讯。对于 1, 3 节点 3 次通讯都成功，因此他们的 PS 是 100%，而节点 2，只有两次成功，因此它的 PS 是 $2/3=66.7\%$ 。显然 PS 是用于评估节点的指标，减小重传可有效提升 PS，实际应用中客户一般要求在 10000 次电芯数据采集时，PS 需大于 85%。

3 TI WBMS SDK 统计命令集

TI 最新的 WBMS SDK 已在底层实现统计功能，用户可以通过 Host 命令来获取这些统计信息，请通过官网下载 STACK，并仔细阅读，如下是统计功能文档在 SDK 中的位置。

```
x:/ti/simplelink_wbms_sdk_x_xx_xx_xx/docs/wbms/doxygen/html/group__STATISTICS__COMMANDS.html
```

3.1 统计命令及其功能 简介

目前支持的统计命令如下所示，具体帧格式请仔细阅读开发文档

- **(0x30) APP_DIAG_GETFWVERSION**
 - 获取 SDK 版本信息
- **(0x32) APP_DIAG_GETNETWORKSTATS**
 - 获取特定节点的延迟和 PER
- **(0x33) APP_DIAG_GETXPACKETS**
 - 获取已成功发送的包数 NumTxSuccessPackets 和发送失败的包数 NumTxFailedPackets。
 - NumTxFailedPackets 是发送失败次数的统计
 - NumTxSuccessPackets 是每次发送成功的统计，重传不计入
- **(0x34) APP_DIAG_GETTHROUGHTPUT**
 - 获取发送和接收的吞吐量，单位是 kbps
- **(0x35) APP_DIAG_GETRXPACKETS**
 - 获取特定节点已成功被 WM 接收的包数 NumRxSuccessPackets 和接收失败的包数 NumRxMissedPackets。
 - NumRxSuccessPackets 是 WM 每次成功接收特定节点包数的统计
 - NumRxMissedPackets 是连续多次重传仍未收到特定节点包数的统计
- **(0x36) APP_DIAG_GETRSSI**
 - 获取特定节点的 RSSI（接收信号强度指示）值
- **(0x37) APP_DIAG_GETJOINPKTCNT**
 - 获取 Join 过程中扫描请求的计数
- **(0x38) APP_DIAG_GETSTATS**
 - 获取 1 个或多个节点的各次重传次数的统计，每个节点会返回 20 个字节，每 4 个字节表示各次重传的次数，最多 5 次重传，需要注意各次重传并不会重复统计，比如某次通讯发生了 2 次重传，则 1 次重传计数不会增加。
 - 获取 0-36 个通讯频道的重传次数统计
- **(0x3D) APP_DIAG_GETKLVDDURATION**
 - 获取 Keep Alive 模式的持续时间（以分钟 为单位）
- **(0x3E) APP_DIAG_GETPARAMS**

- 获取 WBMS 参数如 Dual Main 的角色, WD 的 MAC 地址及状态, 跳频表等

3.2 基于统计命令的 PDR/PS 计算

首先, 是系统 PDR 的计算, 如下公式:

$$PDR_{\text{系统}} = 1 - \frac{\text{Sum}(\text{NumRxMissedPackets } 1 \text{ to } n)}{\text{NumTxSuccessPackets}} \quad (\text{公式 3.2.1})$$

这里, n 是网络中 WD 节点的数量。

即: 累加所有 WD 节点的 NumRxMissedPackets (一次数据采集中 3 次重传没有一次能同时获得所有 WD 的数据), 再除以 NumTxSuccessPackets。实际应用中, 3 次重传过程中各个节点只需要有一次成功上传了采样数据, 可不认为是失败, 因为即使是发生了最大 3 次重传, 各个节点的采样数据时间上的差异仍满足小于 100ms。为此, 额外增加统计参数 **txfail**, 这个参数在每次采样过程中, 在连续 3 次重传后, 一次都没有被成功接收到时才累加 1。按照这一设定, 系统 PDR 可采用如下公式计算获得:

$$PDR_{\text{系统}} = 1 - \frac{\text{txfail}}{\text{NumTxSuccessPackets}} \quad (\text{公式 3.2.2})$$

其次, 是关于各个 WD 节点的 PS, 计算公式如下:

$$PS_{\text{节点}} = 1 - \frac{\text{Sum}(\text{retry } 1 \text{ to } 5) \text{ of Node } n}{\text{NumTxSuccessPackets}} \quad (\text{公式 3.2.3})$$

n 是特定节点, 总的重传次数按如下方式计算:

$$\text{retryTotal} = 1st \text{ retry} + 2nd \text{ retry} \times 2 + 3rd \text{ retry} \times 3 + 4th \text{ retry} \times 4 + 5th \text{ retry} \times 5$$

发生重传即该节点的数据没有被 WM 接收到, 当最大重传次数为 3 时, 4th and 5th retry 计数为 0。

但 PS 的计算需要采用实际从物理层发出的包的数量, 因此, 每次重传时的发送也应该被统计, 而 NumTxSuccessPackets 并不统计重传发生时的发送。比如一次数据采集发送时, 不论经过几次重传, NumTxSuccessPackets 都只增加 1, 因此并不准确。为此, 引入新的统计参数 **txActual**, 该参数是从射频物理层角度统计每一次发送, 如一次采样命令发送过程中, 发生了两次重传, 则该参数计数会增加 3, 即: 第一次发送加上后面两次的重传。因此, 采用如下公式计算节点的 PS:

$$PS_{\text{节点}} = 1 - \frac{\text{Sum}(\text{retry } 1 \text{ to } 5) \text{ of Node } n}{\text{txActual}} \quad (\text{公式 3.2.4})$$

最后，关于各个 WD 节点的 PDR，可使用如下公式计算获得

$$PDR_{\text{节点}} = 1 - \frac{\text{NumRxMissedPackets of Node ID}}{\text{NumTxSuccessPackets}} \quad (\text{公式 3.2.5})$$

显然，如果系统 PDR 可以满足 99.9%的要求，则节点 PDR 一定可以满足。

4 使用 LLCOM 工具测试 WBMS 系统的 PDR、PS

4.1 LLCOM 工具介绍

LLCOM 是一个开源的串口工具，它最主要的特点是支持 Lua 语言扩展，除了具备常用串口工具的串口收发，定时发送，及命令序列运行外，它还可以通过 Lua 脚本解析串口数据并执行自动化操作。常用的串口工具如 WINDOWS 自带的串口调试工具也能支持 JavaScript 脚本扩展，但并不是免费开源软件，LLCOM 代码开源，可以定制出高自由度的串口调试工具。



Figure 1. LLCOM 主界面

接下来，使用 LLCOM 发送 Host 控制命令完成 WBMS 1 对 15 网络的组网。组网完成后，运行 Lua 脚本，自动以设定的间隔时间持续发送电芯电压采样命令。当完成 10000 次采样命令发送后，脚本会自动发送如上描述的各项统计命令，并基于统计命令反馈的结果计算出 PDR、PS 的结果并打印出来。

特别说明：可以使用逻辑分析仪监测 LAUNCHXL-CC26X2R1 的串口，已观察串口数据的收发，确保采样命令的发送间隔小于 100ms，以符合 PDR 测试要求。逻辑分析仪可直接连接 LAUNCHXL-CC26X2R1 的 J1 排针的 DIO2(TX) 和 DIO3(RX)。

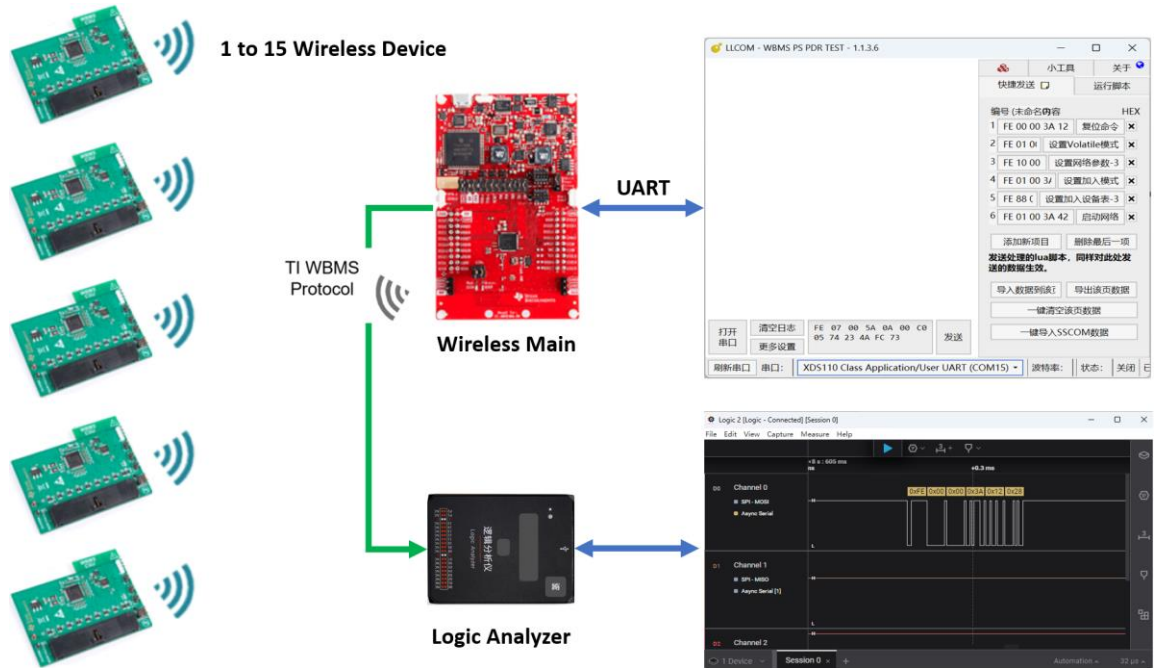


Figure 4. 测试环境搭建

以下是详细步骤，操作视频请点击[链接](#)：

- **Step1** 测试所使用的 WM 和 WD 的固件程序，可直接采用 WBMS SDK 中的默认工程编译。WM 的工程需要在预定义中开启 MANUAL_NETWORK_START，以允许接收来自 Host 的命令来设置启动网络，如下图所示：

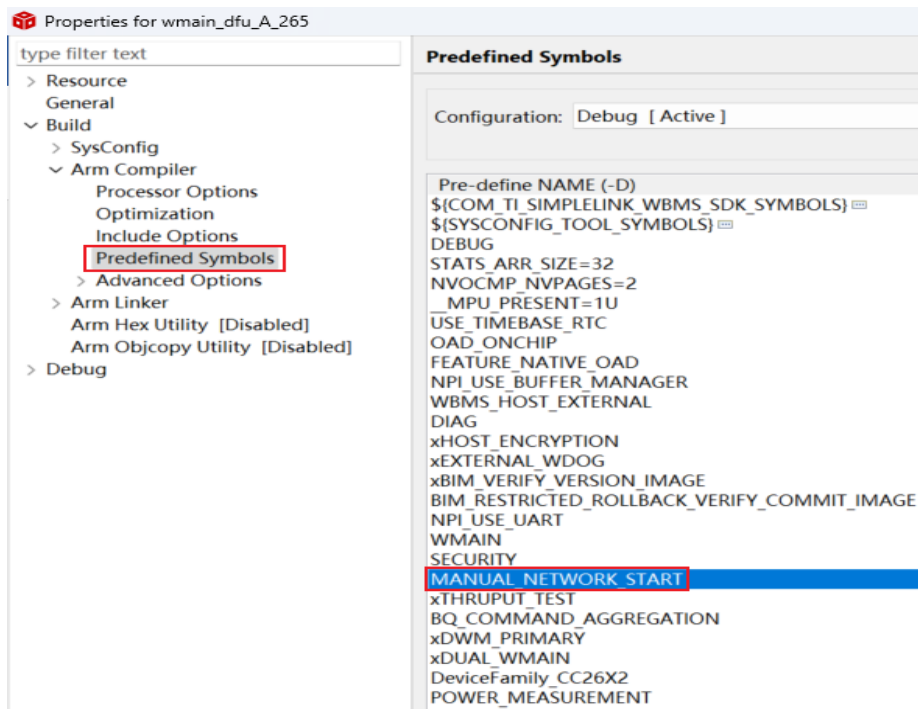


Figure 5. WM 工程中开启 MANUAL_NETWORK_START

如使用的硬件和前文提到的一致，也可从如下链接下载已编译好的固件 bin 文件用于烧录测试。注意虽然有多个 WD 节点，但固件程序使用同一个。固件链接

- **Step 2** 搭建好硬件测试环境后，使用 USB 线连接 [LAUNCHXL-CC26X2R1](#) 和 PC，运行 LLCOM，请通过链接下载这个工具，与本测试相关的命令序列和测试脚本均包含在这个版本的 LLCOM 中。
- **Step 3** LLCOM 中选择正确的串口。通常 LAUNCHXL-CC26X2R1 会显示两个串口，选择“XDS110 Class Application/User UART (COMXX)”，串口波特率选择 921600。

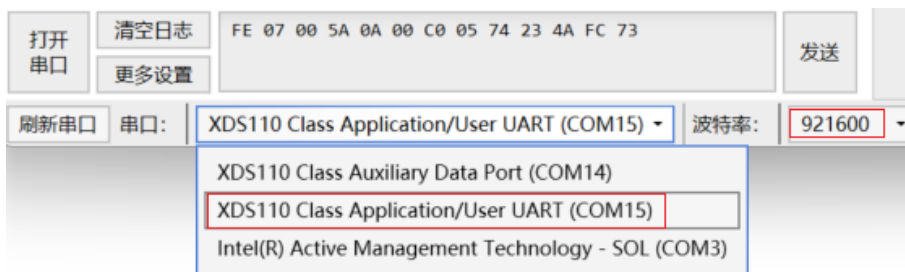


Figure 6. 选择串口及波特率

- **Step 4** 打开串口后，在 LLCOM 的“快捷发送”栏，按顺序 1 - 6 点击发送预设的命令，完成 WBMS 网络的建立。请特别注意点击命令的顺序。如错误，请点击第 1 条复位命令重新开始，关于命令的详细说明请参考 SDK 中的文档。



Figure 7. 快捷发送页面组网命令序列

请注意：命令 3 和命令 5 需要根据测试的实际情况做调整。命令 3 用于设定网络的参数，如网络 ID，WD 节点数量，最大重传次数等。命令 5 则是需要先获得测试所使用的 WD 节点的 MAC 地址。请参考文档 **Network Configuration Commands** 章节的 0x40 和 0x49 命令，文档位置参考如下链接：

simplelink_wbms_sdk_x_y_z/docs/wbms/doxygen/html/group__NETWORK__CONFIGURATION__COMMAND.html

命令 3 示例：FE 10 00 3A 40 00 00 **0F** 00 00 00 00 **03 0E** 00 00 FF FF FF FF FF 97，其中绿色表示 WD 节点数量为 15 个，红色表示最大重传次数为 3 次，蓝色表示 Keep Alive Interval 为 14，这个值需要根据节点数量调整。

命令 5 示例：FE 88 00 3A 49 **0F BF 6B 5B 9B A8 FC FF FF 00 B6 7A 5B 9B A8 FC FF FF 01** 73 6A 5B 9B A8 FC FF FF 02 9A 7B 5B 9B A8 FC FF FF 03 8A 58 5B 9B A8 FC FF FF 04 A2 7B 5B 9B A8 FC FF FF 05 3D 7F 5B 9B A8 FC FF FF 06 F5 7B 5B 9B A8 FC FF FF 07 A5 7B 5B 9B A8 FC FF FF 08 D9 7A 5B 9B A8 FC FF FF 09 C5 6A 5B 9B A8 FC FF FF 0A DF 7A 5B 9B A8 FC FF FF 0B B8 7B 5B 9B A8 FC FF FF 0C A0 69 5B 9B A8 FC FF FF 0D BC 44 5B 9B A8 FC FF FF 0E CD，其中红色是节点数量，绿色为第一个 WD 设备的 MAC 地址，蓝色表示分配的 ID 为 0，其他节点按照同样的 8 字节 MAC 地址和 1 字节分配 ID 格式填入，这里仅用颜色标注了节点 0 和 1。

下图展示了一次成功组网的命令序列及其响应，红色是 LLCOM 发出的命令，绿色则是 WM 的响应。请确保获得正确的响应（“00”表示正确），并且收到的最后一条响应是 **5A 26**。收到 **5A 26** 即表示网络已成功组建。

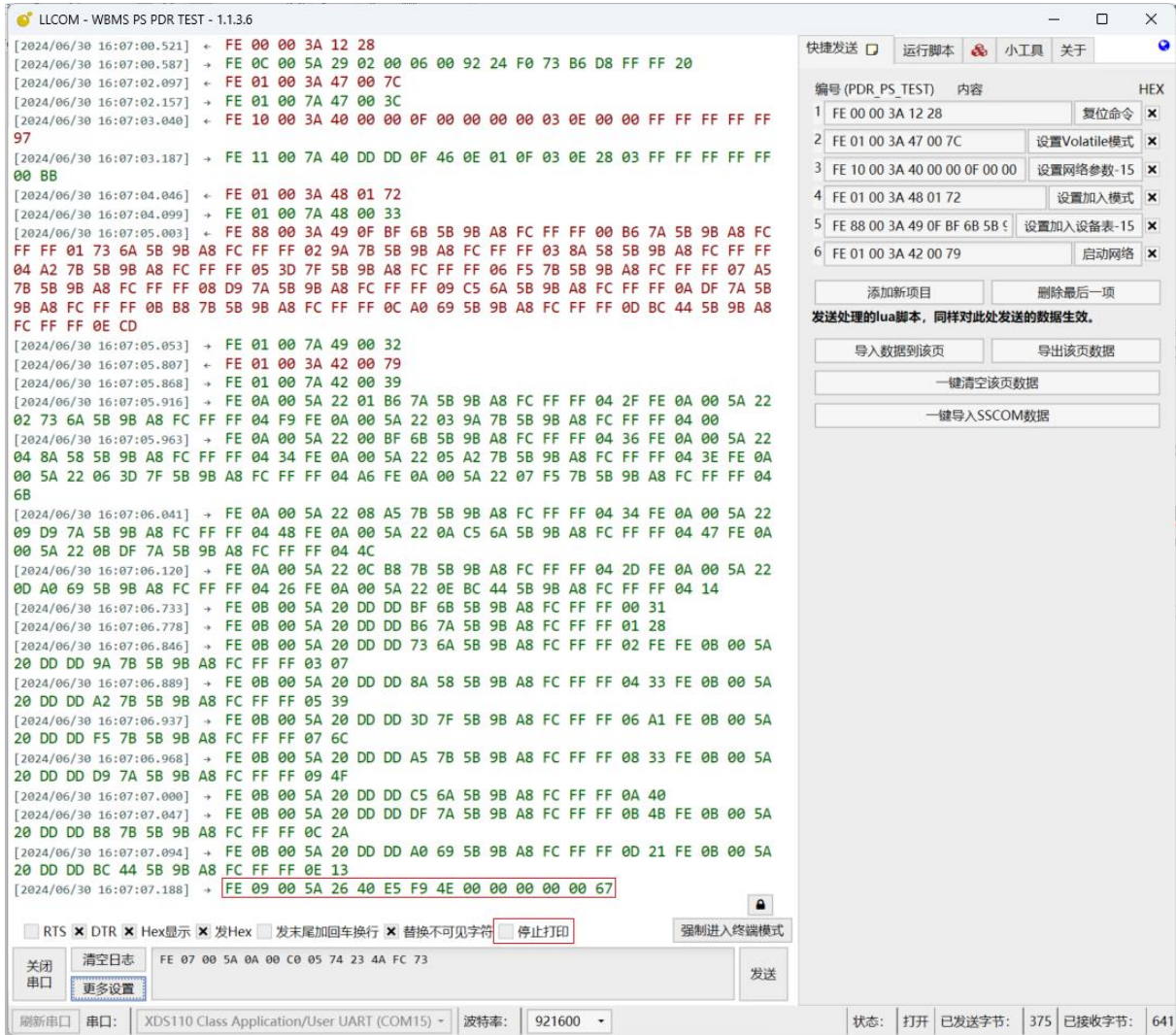


Figure 8. 组网命令序列及其响应

- Step 5** 成功完成组网后，勾选“停止打印”，并进入“运行脚本”页面。选择打开时测试脚本“PSPDR_TEST.lua”，测试代码全部可见。请根据网络WD节点数量和统计结果打印频率，修改cmdBQvoltInterval和cmdTestCycle参数。cmdBQvoltInterval设置为70，可以确保发送命令间隔小于100ms（参见Figure 10），cmdTestCycle实际测试时可以设置为10000。en_print_GETSTATSE和en_print_GERXPACKETS设置是否开启打印额外的统计信息，默认关闭。



Figure 9. 选择测试脚本

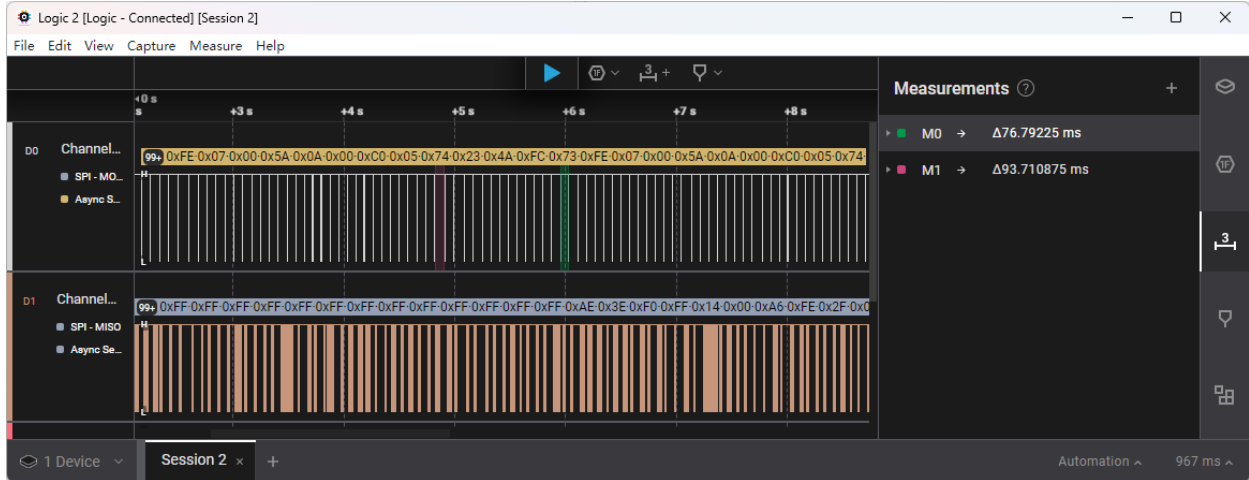


Figure 10. 逻辑分析仪测量通信间隔小于 100ms

- **Step 6** 点击绿色三角箭头，运行脚本，开始测试。当发送次数达到设定的 cmdTestCycle 后，会自动打印统计结果，如下是一次发送了 10000 包数据后的统计信息：

NumTxSuccessPackets: 10707, NumTxFailedPackets: 0, TotalNumRxMissedPackets: 1

txActual: 12682, txfail: 1

System PDR: 99.99%, actual PDR 99.99%

node PDR:0(100.0%)-1(100.0%)-2(100.0%)-3(100.0%)-4(100.0%)-5(100.0%)-6(100.0%)-7(100.0%)-8(100.0%)-9(100.0%)-10(100.0%)-11(100.0%)-12(100.0%)-13(100.0%)-14(100.0%)

node PS:0(99.3%)-1(99.1%)-2(97.7%)-3(96.9%)-4(99.7%)-5(99.3%)-6(98.9%)-7(96.9%)-8(96.1%)-9(97.1%)-10(97.2%)-11(95.8%)-12(96.4%)-13(94.3%)-14(95.6%)

此打印信息显示：

成功发送了 10707 包，发送失败 0 次，总的接收失败是 1 次

txActual 为统计了重传的发送次数，为 12682，txfail 为 1

System PDR 采用（公式 3.2.1）计算而来，而 actual PDR 则基于（公式 3.2.2）

node PDR 基于（公式 3.2.5），node PS 则基于（公式 3.2.4）

node PDR 和 PS 的打印格式如 2(98.3%) 表示节点 2，括号内的是对应的计算结果。

请参考前面 3.2 节的描述。

另外，如果出现一次通讯中多次重传仍失败时，会打印如下信息：

23 fail: 23 01 2

Retry Failed: 0(4)-1(4)-2(4)-3(4)-4(4)-5(4)-6(4)-7(3)-8(3)-9(3)-10(4)-11(0)-12(4)-13(3)-14(3)

这条信息显示，11号节点在这次数据采集通讯中，数据一次都没有被 WM 接收到。

最后，如果在 Lua 脚本中将 **en_print_GETSTATSE** 和 **en_print_GERXPACKETS** 设置为 1，则会打印出如下额外的信息，这些信息将有助于分析各个节点在各次重传次数、NumRxMissedPackets 及 37 个无线频道重传次数的信息，如下数据所示：

Retry Node n = [r1:1], [r2:0], [r3:0], [r4:0], [r5:0]，其中 r1 表示 1 次重传

Rx Node n = Success: 101, Missed:0Channel 0 = 0

Channel n = 1

4.3 LLCOM PDR PS 测试的总结

LLCOM 支持 Lua 语言扩展，不到 300 行代码即可完成对 WBMS 系统的 PDR 及 PS 的测试。

5 参考文献

1. [CC2662R-Q1 SimpleLink™ 无线 BMS MCU 数据表 \(Rev. C\)](#)
2. [BQ79616-Q1 数据手册](#)
3. [SIMPLELINK-WBMS-SDK Document and User guide](#)
4. [BQ79616-Q1 软件设计参考](#)
5. [The next generation Wireless BMS using the CC2662R-Q1](#)
6. [More miles, less wires: Revolutionizing automotive battery management](#)
7. [Using wireless technologies to replace cables in car access and battery management](#)

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2024，德州仪器 (TI) 公司