

适用于 Jacinto™7 系列的 LPDDR 4 电子眼图工具

Biao Li (李彪), Harry Zhao (赵海仁), Joe Shen (申颖)

Central FAE

摘要

Jacinto™ 7 系列处理器是 TI 公司基于 Keystone 架构推出的最新一代汽车处理器，在实际的应用中需要搭配使用 LPDDR4 高性能低功耗内存。随着大量车型的量产上市，客户不可避免会遇到一些随机错误，无法判断是否由于 LPDDR4 的信号质量原因导致，在实际的量产项目中，由于硬件设计和一些量产保护措施的实施，使得对 LPDDR4 的信号质量的测试变得困难，无法使用传统的示波器眼图来衡量 LPDDR4 的信号质量，因此 TI 提供了电子眼图工具以针对此类情况，方便对 LPDDR4 的信号质量进行一个粗略的测量并提供参考。

本文将针对电子眼图的使用以及判断标准对本工具进行详细介绍，本工具无法作为量产工具使用，仅供调试参考使用。本文涉及到的工具包，请联系您的 TI 技术支持或者 TI 的销售代表。

本工具输出的结果仅能代表测量时刻的 LPDDR4 信号质量，无法显示整个运行过程中的状态。本工具仅适用于 Jacinto7 系列，TI 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、Web 工具、安全信息和其他资源（不保证任何错误），这些资源面向使用 TI 产品进行设计的熟练开发人员。您全权负责为您的应用选择适当的 TI 产品，设计、验证和测试您的应用，以及确保您的应用符合适用标准以及任何其他安全其他要求。这些资源如有更改，恕不另行通知。TI 授予您使用这些资源的权限，仅用于开发使用资源中描述的 TI 产品的应用程序。禁止以其他方式复制和展示这些资源。

目录

1. 电子眼图工具介绍	3
1.1 电子眼图工具组成	3
1.2 电子眼图的基本工作原理	3
2. 电子眼图工具的使用	3
2.1. 通用（GP）芯片上电子眼图的运行	3
2.2. 高安芯片上电子眼图的使用	6
3. 电子眼图结果判断	7
3.1. 电子眼图结果介绍	7
3.2. 读眼图结果判断	8
3.3. 写眼图结果判断	9
4. 在电子眼图使用中的常见问题	11
4.1. 打印串口的更改	11
4.2. Perl 脚本的运行	12
5. 总结	13
6. 参考文献	13

1. 电子眼图工具介绍

电子眼图工具是由 TI Jacinto 产品线硬件应用工程师团队开发的针对 Jacinto7 系列芯片的专用工具，目前的版本仅支持 TDA4VM、DRA821 芯片，后续对 Jacinto7 系列的其他产品也会陆续进行支持。电子眼图工具主要针对的是客户板卡中与 DDR 强相关的问题，例如 DDR memory test 出现错误，芯片启动异常，启动成功后随机死机等可能和 DDR 信号质量有关系的问题。在以往的调试经验中，从硬件角度去判断 DDR 的数据线以及时钟线的信号质量需要使用高速示波器对 DDR 进行眼图的测量，但是在实际的工程应用中，客户面临示波器眼图测量受限的情况，因此 TI 提供了该工具，对 DDR 的信号质量在软件层面进行一个粗略的评估。

1.1 电子眼图工具组成

“bin”文件夹中包含预构建的可执行二进制文件，“build”文件夹中包含 CCS 的项目文件，“doc”文件夹中包含简要的应用手册和“example”文件夹。“example”文件夹中的示例，包括：在 DUT 上运行的二进制可执行文件的输出，以及后处理脚本的输出，“gel”文件夹中包含 GEL 文件用于在 CCS 中导入，作用是将 PINMUX 配置为 UART IO，并且配置 R5 RAT，“lib”文件夹中包含源代码中所使用到的库文件和头文件，“perl”文件夹中包含后处理应用程序，在“src”文件夹中包含 DUT 可执行文件的源代码（不包括库）。

电子眼图工具目前能够支持的功能如下：能显示 WRITE 和 READ DQ 数据线位的二维眼图图像。垂直坐标表示参考电压，水平坐标表示运行时间步长的延迟值。电子眼图工具会对每个 DQ 位都提供的二维眼图图像。并且为 DQS 的两个边缘提供绘图，并且使用颜色编码绘图来说明：训练值（紫色方块）、穿越区域（绿色区域）、代表故障率的渐变颜色（由红到黄）。支持自动检测总线宽度以优化模式生成，支持自定义运行时间步长（电压和延迟）以实现客户使用的灵活性，支持客户通过选择不同的运行时间步长可以达到缩短测试时间或者提高测试精度的目的。电子眼图工具是通过 UART 串口进行输出测试结果。

当前电子眼图工具也存在一定的局限性：每次测试只能测试读取或者写入的电子眼图，执行读取或者写入之后需要对 DDRSS 进行重新配置；当前工具以原始值而非实际电压值或者近似时间值来显示坐标；参考电压（VREF）的最小步长限制为 2，延迟最小步长限制为 8；需要使用 CCS（Code Composer Studio），需要使用 JTAG 调试接口；后处理脚本仅在 Windows 操作系统上测试完成（目前仅支持 Windows）；提供的 GEL 文件目前仅支持 UART0（EVM 默认串口）输出，需要在其他串口打印需要更改 PINMUX。

1.2 电子眼图的基本工作原理

电子眼图工具对每条数据线进行读或写操作，根据客户输入的参考电压和测试时间步长，并且实时校验对应操作的正确和错误率，对每条数据线进行逐一测试后，生成对应的错误字节数量，错误的比例代表该数据线上，该参考电压下的信号质量（Margin）的情况，根据该数量对对应方块进行填色，最后生成类似眼图的二维图像。其中紫色方块代表训练值位置，该值应该处于绿色方块靠近居中位置。

2. 电子眼图工具的使用

2.1. 通用（GP）芯片上电子眼图的运行

在通用芯片上运行电子眼图，由于 DMSC（M3 CORE）和 MCU1_0 等核的 JTAG 的访问权限没有被闭锁，因此在 GP 芯片上相对 HS 板卡更加灵活，主要分为以下几个步骤，下列步骤通用于 TDA4VM 和 DRA821：

1. Bootmode 更改到串口启动模式，并连接好 JTAG 调试口，EVM 设置和客户板卡 bootmode 设置均可参考 [EVM Setup for J721E](#)；

2. 加载 TDA4VM/ DRA821 的 targetconfiguration 文件，详细步骤可参考 [CCS Setup for J721E](#);
3. 连接到 M3 核后，首先需要进行配置 LPDDR4，通过加载 J7ES_LPDDR4_Config 文件（路径在 CCS 界面上 Scripts > J7ES DDR Memory config 下）来实现，客户板卡如更换 DDR 颗粒，则需要通过 [配置表格](#) 在 gel 界面生成后进行替换；

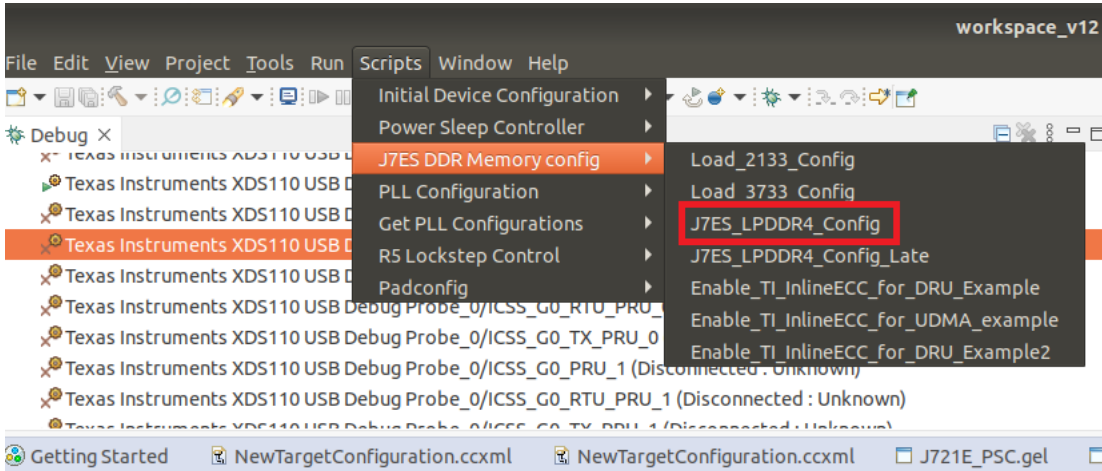


图 1 连接到 M3 后运行 J7ES_LPDDR4_Config

4. 之后需要对 R5F 的 RAT 进行配置，在 CCS 中可以通过加载“Configure_RAT”（路径在 CCS 界面上 Scripts > Initial Device Configuration 下）脚本来实现，并且将 Initial Device Configuration 目录下的“Configure_ATCM”和“Configure_R5F_Bootvectors”都运行一次；

```

DMSC_Cortex_M3_0: GEL Output: DDR PLL Calibration Control MMR value: 0x800001A1
DMSC_Cortex_M3_0: GEL Output: DDR PLL + HSDIV set to 933MHz.
DMSC_Cortex_M3_0: GEL Output: Frequency change request type 2 received from controller
DMSC_Cortex_M3_0: GEL Output: Setting DDR PLL + HSDIV to 933MHz
DMSC_Cortex_M3_0: GEL Output: DDR PLL Calibration Control MMR value: 0x80170000
DMSC_Cortex_M3_0: GEL Output: DDR PLL Calibration Control MMR value: 0x80000193
DMSC_Cortex_M3_0: GEL Output: DDR PLL + HSDIV set to 933MHz.
DMSC_Cortex_M3_0: GEL Output: Frequency change request type 1 received from controller
DMSC_Cortex_M3_0: GEL Output: Setting DDR PLL + HSDIV to 933MHz
DMSC_Cortex_M3_0: GEL Output: DDR PLL Calibration Control MMR value: 0x80170000
DMSC_Cortex_M3_0: GEL Output: DDR PLL Calibration Control MMR value: 0x8000016D
DMSC_Cortex_M3_0: GEL Output: DDR PLL + HSDIV set to 933MHz.
DMSC_Cortex_M3_0: GEL Output: Frequency change request type 2 received from controller
DMSC_Cortex_M3_0: GEL Output: Setting DDR PLL + HSDIV to 933MHz
DMSC_Cortex_M3_0: GEL Output: DDR PLL Calibration Control MMR value: 0x80170000
DMSC_Cortex_M3_0: GEL Output: DDR PLL Calibration Control MMR value: 0x8000019E
DMSC_Cortex_M3_0: GEL Output: DDR PLL + HSDIV set to 933MHz.
DMSC_Cortex_M3_0: GEL Output: --->>> All frequency change requests have completed... <<<---
DMSC_Cortex_M3_0: GEL Output: --->>> LPDDR4 Initialization is DONE! <<<---
DMSC_Cortex_M3_0: GEL Output: This GEL is currently only supported for use from the Cortex-M3 inside the DMSC.
DMSC_Cortex_M3_0: GEL Output: Do not run this GEL from any other CPU on the SoC.
DMSC_Cortex_M3_0: GEL Output: This script sets the first address translation region to [0x8000_0000, 0x0000_0000].
DMSC_Cortex_M3_0: GEL Output: It also sets the second address translation region to [0x6000_0000, 0x4000_0000].
DMSC_Cortex_M3_0: GEL Output: This is consistent with the SoC DV assumptions.
DMSC_Cortex_M3_0: GEL Output: Configuring ATCM for the R5Fs
DMSC_Cortex_M3_0: GEL Output: ATCM Configured.
DMSC_Cortex_M3_0: GEL Output: Configuring bootvectors
DMSC_Cortex_M3_0: GEL Output: Bootvectors configured.

```

图 2 运行成功后的 console 界面

5. 连接到 MCU1_0（在 CCS 中是 MCU R5_0），此时直接加载 bin 文件夹下的.out 文件会显示加载失败，原因是 Beta 版本的工具使用到了 VPAC 部分的地址，但是由于此时 VPAC 还未启动，因此会报错无法访问该地址。目前的 workaround 的方式是先使用 J721E_PSC.gel（路径在 ~/ti/ccs1220/ccs/ccs_base/emulation/gel/J721E_DRA829_TDA4VM/gel/J721E_PSC/J721E_PSC.gel）文件对 VPAC 进行上电，并且需要使用电子眼图工具中 gel 文件夹下的 Jacinto7_DDRSS_Tools_R5.gel 文件对 Uart 口等进行配置，之后必须再对 MCU R5_0 进行一次 CPU Reset，之后再加载 bin 文件夹下的固件（jacinto7_ddrssi_tools_r5.out），最后点击运行后会在对应的串口打印 log。

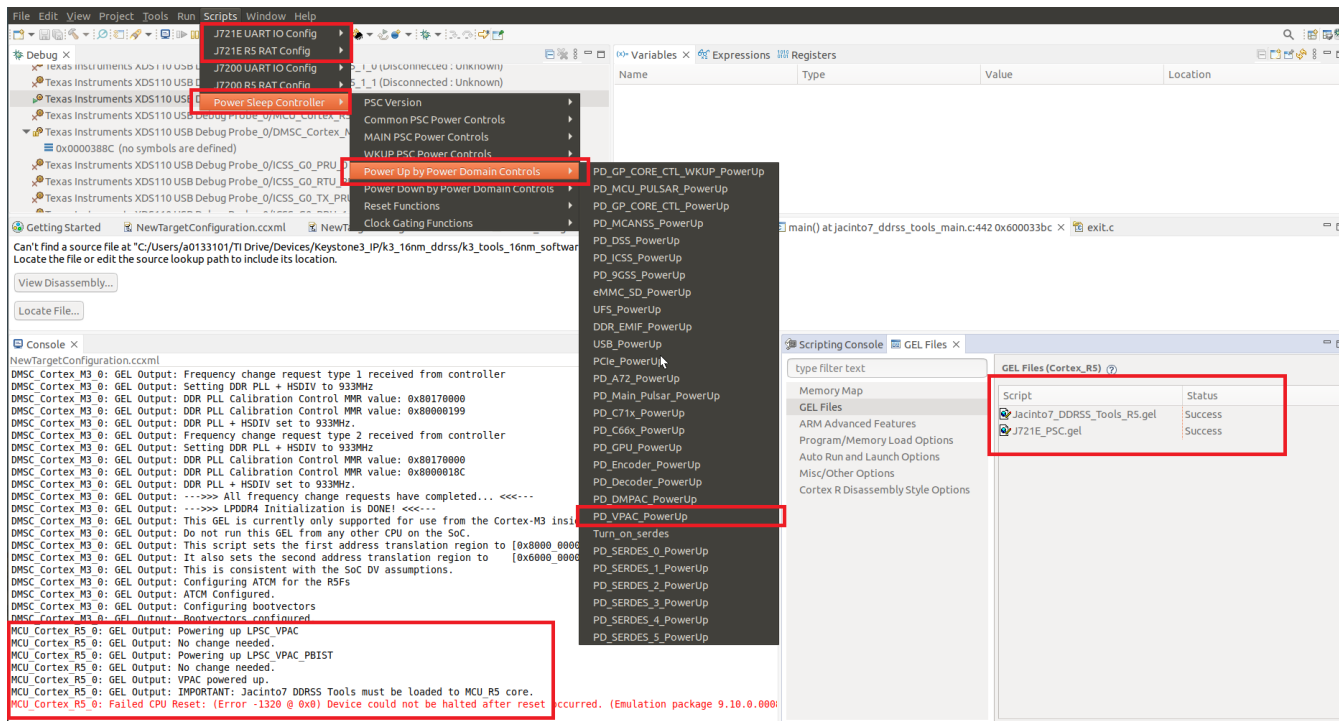


图 3 连接到 R5_0 后使用 gel 脚本进行配置

6. 运行 R5 后，会在配置的串口出现打印，此时会提示客户输入相关的参数：
 - a. 选择输出“读”（0）或“写”（1）的电子眼图；
 - b. 选择 VREF 的步长（最小为 2）；
 - c. 选择时间延迟的步长（最小为 8）；

```

Jacinto7 DDRSS Tools
Supported SOCs: J721E, J7200
Revision: Beta Release - September 2020

DDRSS Instance: 0
  DRAM Type: LPDDR4
  Bus Width: 32
  Rank Count: 2

Margin Analysis Input:
  Eye Type (Select '0' for read, '1' for write):
  Eye Precision (VREF / DLL Step Size; impacts test time):
    VREF Step Size (enter decimal value; min value is 2):
    Delay Step Size (enter decimal value; min value is 8):
  Pattern: wc_isi_sso
  Transfer Size (bytes): 487872
eye_type=write
mr14_f0_cs0: 0x0
mr14_f0_cs1: 0x0
mr14_f0_cs2: 0x0
mr14_f0_cs3: 0x0
    
```

图 4 串口打印示例

7. 保存串口的全部的 log 在 xxx.txt 中（任何英文名的空白文本文件中，如 LP4_Read_DlyStep16_VrefStep4_Example.txt）；
8. 使用预处理脚本对得到的 log 进行处理。目前该脚本仅支持在 Windows 的命令行中运行。预处理的时间取决于设置的延迟时间的步长，选择的步长越小，处理时间将越长，在 Windows 命令行中运行需要有 Perl 的

环境，将得到的 log 文件拷贝到电子眼图工具的 perl 文件夹中，使用以下命令可以在输入（perl）文件夹下生成对应的 excel 表格形式的电子眼图，例：cd 到 perl 文件夹下后执行以下指令：

jacinto7_ddrss_tools_margin_Analysis.pl LP4_Read_DlyStep16_VrefStep4_Example.txt, 会在 txt 文件所在文件夹下生成名为 LP4_Read_DlyStep16_VrefStep4_Example.xlsx 文件。使用 Excel 打开之后，可以看到如下的眼图示意图。

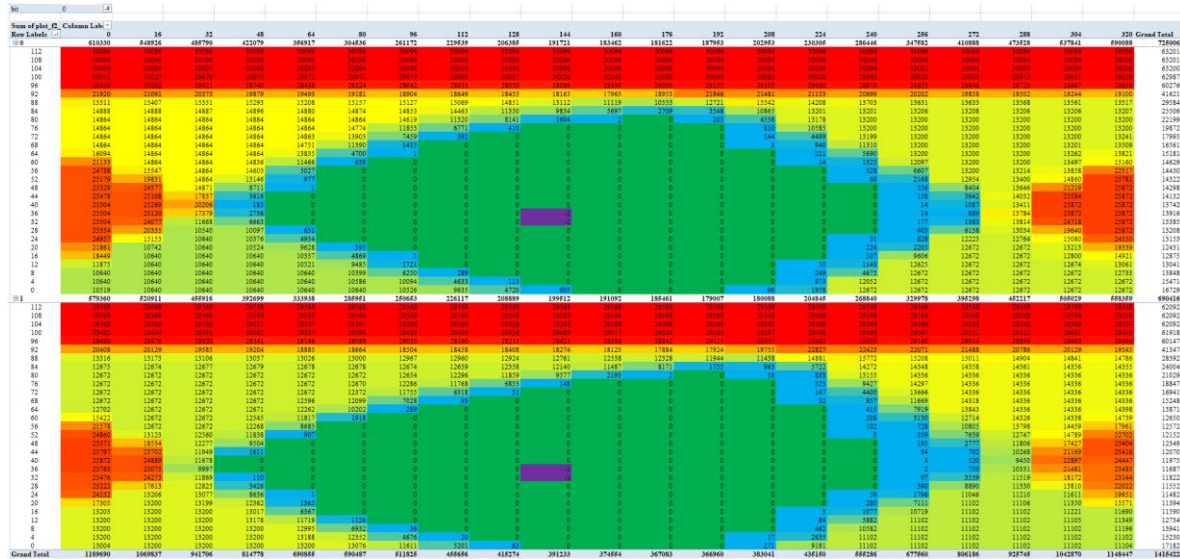


图 4 电子眼图示例

2.2. 高安芯片上电子眼图的使用

Jacinto™ 7 系列处理器根据信息安全的要求不同可以分为通用（GP）芯片和高安（HS）芯片，HS 芯片中，加入了 OTP 标识 eFuse 以及多个硬件安全加速器，增加了芯片在启动过程中对系统镜像的加解密以及签名验证，能够避免系统遭受外围的恶意篡改，删除以及复制。TI 提供 HS 的芯片给客户以进行量产，HS 芯片目前正在成为市面上量产车型的主流，在 HS 芯片中的 JTAG 作为开发过程以及未来售后都必要的外部接口，一般会存在产品的整个生命周期。所以 JTAG 作为唯一必要的对外接口，其安全性具有至关重要的地位，因此在 HS-SE（烧录了客户密钥）的芯片中，已经将 M3 核（DMSC）的 JTAG 口进行了永久闭锁，如表 1 所示。为了防止外部通过 JTAG 口对客户的密钥进行读取，造成泄露的风险。除此之外，其他核的 JTAG 口也进行了暂时性的闭锁，而电子眼图工具强依赖于 JTAG 口进行程序的运行和 DDR 的配置等工作。

表 1 不同芯片类型中的 JTAG 权限及状态

芯片类型	芯片状态	M3 JTAG 状态	其它核心 JTAG 状态
General Purpose (GP)	GP	Open	Open
High Security (HS)	HS-FS	Closed	Open
High Security (HS)	HS-SE	Closed	Controlled

因此在高安芯片上运行电子眼图相对于通用芯片在前五个步骤上需要进行一些改动。

1. 首先需要准备一张在待测试板上能够正常启动并且能进入 Linux 系统的 SD 卡，并将板卡设置为 SD 卡 boot 模式；
2. 在 SBL 中对程序的修改，将以下修改更新到 sbl_main.c 中，重新编译后替换 SBL。主要的修改是增加了一个死循环，让启动流程卡在 SBL 阶段。由于 HS 芯片的 M3 核永久闭锁，无法使用 M3 核来执行 DDR 配置等操作，因此需要使用客户的程序进行配置，并且在配置完成后，卡在 SBL 阶段，再进行后续的流程；

```
diff --git a/packages/ti/boot/sbl/board/k3/sbl_main.c b/packages/ti/boot/sbl/board/k3/sbl_main.c
index dc1fc6d..d9f10aa 100755
--- a/packages/ti/boot/sbl/board/k3/sbl_main.c
+++ b/packages/ti/boot/sbl/board/k3/sbl_main.c
@@ -327,6 +327,15 @@ const CSL_ArmR5MpuRegionCfg gCslR5MpuCfg[CSL_ARM_R5F_MPU_REGIONS_MAX] =
};

#endif
+
+void StartupEmulatorWaitFxn (void)
+{
+ volatile uint32_t enableDebug = 1;
+ do
+ {
+ }while (enableDebug);
+}
+
int main()
{
#ifdef SBL_ENABLE_HLOS_BOOT) && (defined(SOC_J721E) || defined(SOC_J7200)
@@ -457,7 +466,7 @@ int main()
Board_init(BOARD_INIT_DDR);
SBL_log(SBL_LOG_MAX, "done.\n");
#endif
-
+StartupEmulatorWaitFxn();
#ifdef SBL_USE_MCU_DOMAIN_ONLY) && !defined(SBL_ENABLE_DEV_GRP_MCU)
/* Enable GTC */
SBL_log(SBL_LOG_MAX, "Initializing GTC ...");
```

3. 程序执行到 SBL 阶段后，tifs.bin 已经将 R5_0 的 JTAG 口进行了解锁，此时连接到 MCU R5_0，与通用芯片类似进行加载两个 gel 文件：J721E_PSC.gel（路径在 ~/ti/ccs1220/ccs/ccs_base/emulation/gel/J721E_DRA829_TDA4VM/gel/J721E_PSC/J721E_PSC.gel）；gel 文件夹下的 Jacinto7_DDRSS_Tools_R5.gel（~/jacinto7_ddrssi_tools_beta/gel/Jacinto7_DDRSS_Tools_R5.gel），必须先对 MCU R5_0 进行一次 CPU Reset，再执行对 VPAC 上电和串口配置等操作（与通用芯片的操作不同）；
4. 加载 bin 文件夹下的固件（~/jacinto7_ddrssi_tools_beta/bin/jacinto7_ddrssi_tools_r5.out），最后点击运行后，会在配置对应的串口（默认是 Uart0）打印 log。
5. 后续步骤与通用（GP）芯片类似，这里不在赘述。

3. 电子眼图结果判断

3.1. 电子眼图结果介绍

电子眼图的生成方法为通过在一定范围内不同 VREF 和时间延迟下进行 DDR 读写测试，并记录读写测试失败的次数，最终呈现在 Excel 表格中。下图为在 Excel 表格中生成的电子眼图示意图，最直观可以看到的是整个表格由不同的颜色组成，其中不同的颜色代表不同的读写测试失败次数，在每一个方格中的数字代表失败的次数。紫色区域代表最终 training 的值，绿色区域代表所有读写测试均通过，最终眼高和眼宽的判断也是根据这一区域的大小；其他颜色（蓝色、黄色、橙色、红色）代表读写测试中出现错误，并且颜色随着出现错误数量的增多而加深。

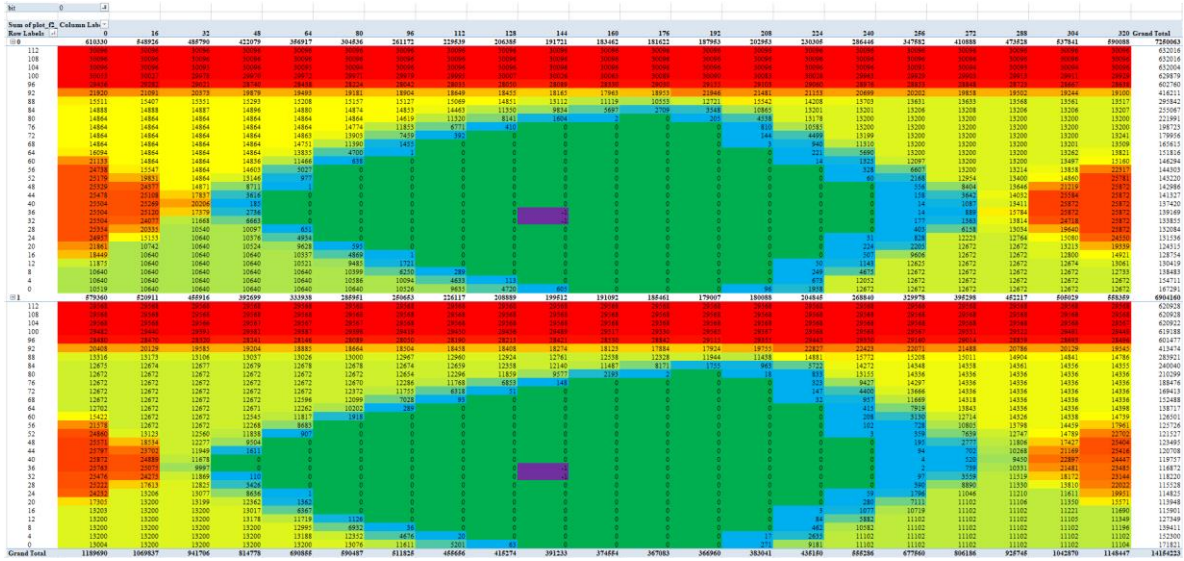


图 5 电子眼图示例

在电子眼图中，最上方和最左侧各有一系列数字，在以下的介绍中将其称为横轴和纵轴，其中横轴代表时间延迟，纵轴代表 VREF。其中每一个方格所代表的范围由 2.1 章节第 6 步中设置的时间延迟和 VREF 的步长决定。在读眼图和写眼图中，横纵轴对应值的范围不同，详细介绍请参考 3.2 和 3.3 章节。

更多详细 LPDDR4 设计相关信息请参考以下文档：

[Jacinto 7 LPDDR4 Board Design and Layout Guidelines \(Rev. E\)](#)

3.2. 读眼图结果判断

如下图所示，读眼图对于眼高和眼宽的判断为钻石型，对于不同的 LPDDR4 频率，对于 VREF（眼高）和时间延迟（眼宽）的要求有一些区别，需要根据实际使用的 LPDDR4 频率进行选择，以下所有计算以 LPDDR4-4266 频率为准进行介绍。

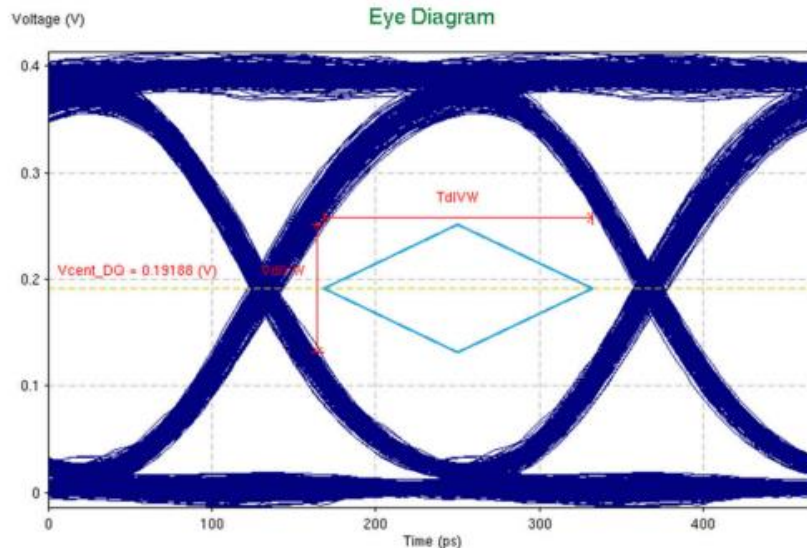


Figure 3-5. Sample Simulated LPDDR4-4266 Read Eye With Diamond-Shaped Eye Mask

Table 3-3. LPDDR4/4x Eye Mask Definitions/Requirements

Parameter	Mask Shape	LPDDR4-3200	LPDDR4-3733	LPDDR4-4266
CA eye mask TcIVW	Rectangular (1)	0.3 UI (1)	(2)	0.3 UI (1)
CA eye mask VcIVW	Rectangular (1)	155 mV (1)	(2)	145 mV (1)
Write eye mask TdIVW	Rectangular (1)	0.25 UI (1)	(2)	0.25 UI (1)
Write eye mask VdIVW	Rectangular (1)	140 mV (1)	(2)	120 mV (1)
Read eye mask TdIVW	Diamond	0.61 UI	0.66 UI	0.7 UI
Read eye mask VdIVW	Diamond	140 mV	120 mV	120 mV

- (1) Copied from JEDEC specification: Low Power Double Data Rate 4 (LPDDR4).
- (2) For details, contact the DRAM vendor.

图 6 读眼图判断标准

以下图 7 为读眼图示例，其中横轴（时间延迟）范围为 0 到 320，每一个值代表 1/512 tck，纵轴（VREF）范围为 0 到 112，代表 5.4% VDDQ 到 25.9% VDDQ，每一个值大约代表 0.183% VDDQ。其中 VDDQ 为 1.1V，即每一个值大约代表 2mV。

在图 6 的读眼图判断标准中，对于时间延迟要求为 0.7UI，其中 1UI 代表 1/2tck，即对时间延时宽度的要求为 0.35tck，即可根据横轴每一个值代表的大小最终计算出眼宽对应的单元格数量。对于 VREF 的要求为 120mV，即可根据纵轴每一个值代表的大小最终计算出眼高对应的单元格数量。以 2.1 章节第 6 步中时间延时步长设置为 16，VREF 步长设置为 4 为例，眼宽需要满足 11.2 个单元格的宽度，眼高需要满足 15 个单元格的高度，以眼高和眼宽为基准在 DDR 数字眼图结果中构建钻石型，如下图紫色框所示，需要满足该钻石型部分未超出绿色区域的范围。

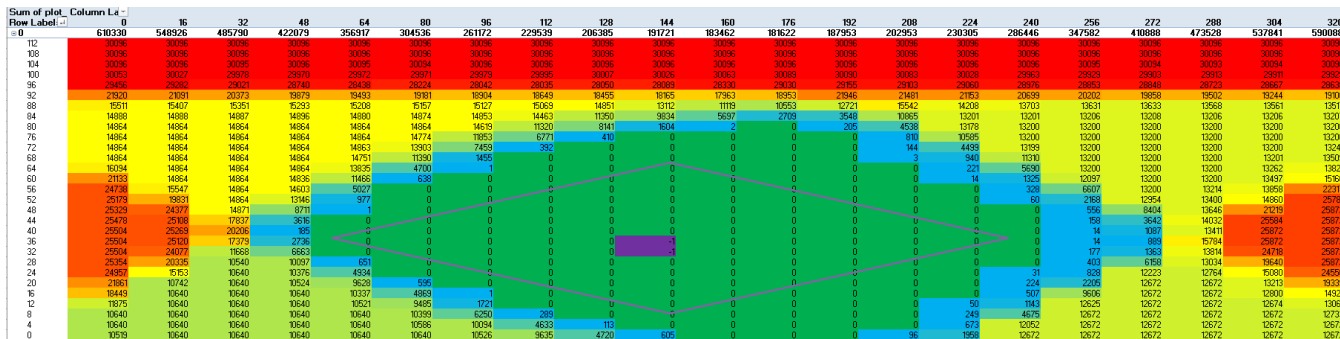


图 7 读眼图示例

3.3. 写眼图结果判断

如下图所示，写眼图对于眼高和眼宽的判断为矩形，对于不同的 LPDDR4 频率，对于 VREF（眼高）和时间延迟（眼宽）的要求有一些区别，需要根据实际使用的 LPDDR4 频率进行选择，以下所有计算以 LPDDR4-4266 频率为准进行介绍。

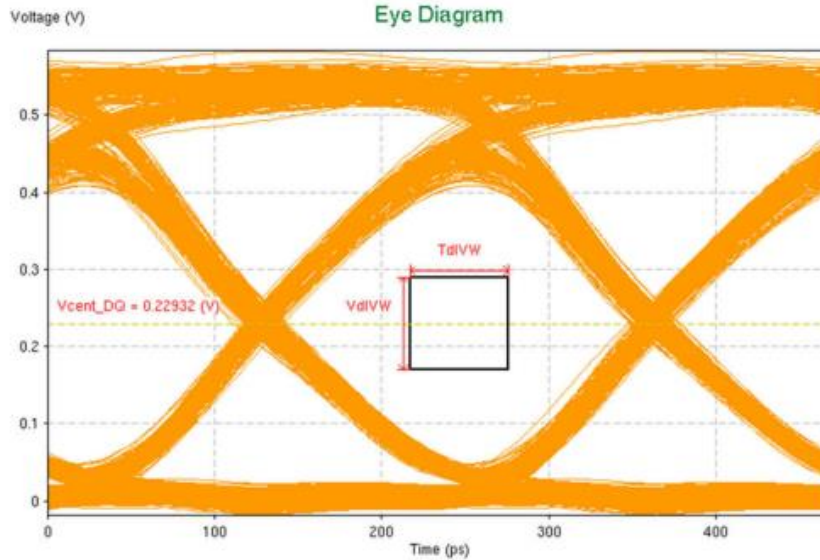


Figure 3-6. Sample Simulated LPDDR4-4266 Write Eye With Rectangular JEDEC Eye Mask

Table 3-3. LPDDR4/4x Eye Mask Definitions/Requirements

Parameter	Mask Shape	LPDDR4-3200	LPDDR4-3733	LPDDR4-4266
CA eye mask TcIVW	Rectangular ⁽¹⁾	0.3 UI ⁽¹⁾	⁽²⁾	0.3 UI ⁽¹⁾
CA eye mask VcIVW	Rectangular ⁽¹⁾	155 mV ⁽¹⁾	⁽²⁾	145 mV ⁽¹⁾
Write eye mask TdIVW	Rectangular ⁽¹⁾	0.25 UI ⁽¹⁾	⁽²⁾	0.25 UI ⁽¹⁾
Write eye mask VdIVW	Rectangular ⁽¹⁾	140 mV ⁽¹⁾	⁽²⁾	120 mV ⁽¹⁾
Read eye mask TdIVW	Diamond	0.61 UI	0.66 UI	0.7 UI
Read eye mask VdIVW	Diamond	140 mV	120 mV	120 mV

(1) Copied from JEDEC specification: Low Power Double Date Rate 4 (LPDDR4).
 (2) For details, contact the DRAM vendor.

图 8 写眼图判断标准

以下图 9 为写眼图示例，其中横轴（时间延迟）范围大约为 810 到 1300，每一个值代表 1/512 tck，纵轴（VREF）范围为 0 到 72，其代表的 VDDQ 范围参考图 10 MR14 寄存器对照表所示，0 到 50 代表 Range[0]，从 10% 到 30% VDDQ，51 到 72 代表 Range[1]并跳过 Range[0]范围的重合部分，即 51 对应 30.4% VDDQ，并依此类推，每一个值代表 0.4% VDDQ。其中 VDDQ 为 1.1V，即每一个值代表 4.4mV。

在图 9 的写眼图判断标准中，对于时间延迟要求为 0.25UI，其中 1UI 代表 1/2tck，即对时间延时宽度的要求为 0.125tck，即可根据横轴每一个值代表的大小最终计算出眼宽对应的单元格数量。对于 VREF 的要求为 120mV，即可根据纵轴每一个值代表的大小最终计算出眼高对应的单元格数量。以 2.1 章节第 6 步中时间延时步长设置为 16，VREF 步长设置为 4 为例，眼宽需要满足 4 个单元格的宽度，眼高需要满足 7 个单元格的高度，以眼高和眼宽为基准在 DDR 数字眼图结果中构建矩形，如下图紫色框所示，需要满足该矩形部分未超出绿色区域的范围。

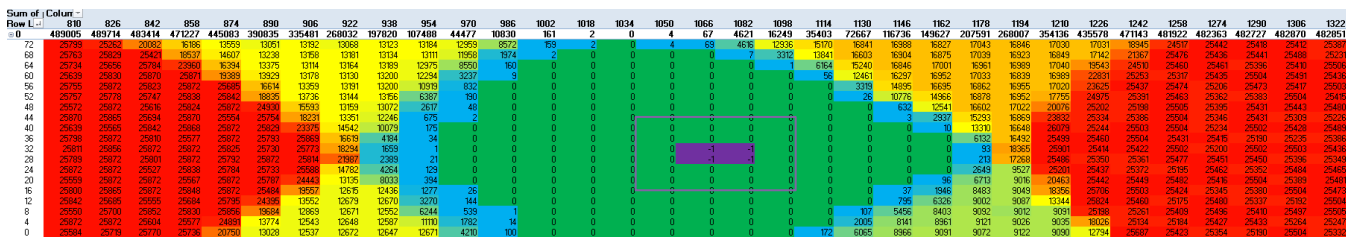


图 9 写眼图示例

Function	Operand	Range[0] Values (% of V _{DDQ})	Range[1] Values (% of V _{DDQ})	Notes	
V _{REF} Settings for MR14	OP[5:0]	000000 _B : 10.0%	011010 _B : 20.4%	000000 _B : 22.0%	011010 _B : 32.4%
		000001 _B : 10.4%	011011 _B : 20.8%	000001 _B : 22.4%	011011 _B : 32.8%
		000010 _B : 10.8%	011100 _B : 21.2%	000010 _B : 22.8%	011100 _B : 33.2%
		000011 _B : 11.2%	011101 _B : 21.6%	000011 _B : 23.2%	011101 _B : 33.6%
		000100 _B : 11.6%	011110 _B : 22.0%	000100 _B : 23.6%	011110 _B : 34.0%
		000101 _B : 12.0%	011111 _B : 22.4%	000101 _B : 24.0%	011111 _B : 34.4%
		000110 _B : 12.4%	100000 _B : 22.8%	000110 _B : 24.4%	100000 _B : 34.8%
		000111 _B : 12.8%	100001 _B : 23.2%	000111 _B : 24.8%	100001 _B : 35.2%
		001000 _B : 13.2%	100010 _B : 23.6%	001000 _B : 25.2%	100010 _B : 35.6%
		001001 _B : 13.6%	100011 _B : 24.0%	001001 _B : 25.6%	100011 _B : 36.0%
		001010 _B : 14.0%	100100 _B : 24.4%	001010 _B : 26.0%	100100 _B : 36.4%
		001011 _B : 14.4%	100101 _B : 24.8%	001011 _B : 26.4%	100101 _B : 36.8%
		001100 _B : 14.8%	100110 _B : 25.2%	001100 _B : 26.8%	100110 _B : 37.2%
		001101 _B : 15.2%	100111 _B : 25.6%	001101 _B : 27.2% (Default)	100111 _B : 37.6%
		001110 _B : 15.6%	101000 _B : 26.0%	001110 _B : 27.6%	101000 _B : 38.0%
		001111 _B : 16.0%	101001 _B : 26.4%	001111 _B : 28.0%	101001 _B : 38.4%
		010000 _B : 16.4%	101010 _B : 26.8%	010000 _B : 28.4%	101010 _B : 38.8%
010001 _B : 16.8%	101011 _B : 27.2%	010001 _B : 28.8%	101011 _B : 39.2%		
010010 _B : 17.2%	101100 _B : 27.6%	010010 _B : 29.2%	101100 _B : 39.6%		
010011 _B : 17.6%	101101 _B : 28.0%	010011 _B : 29.6%	101101 _B : 40.0%		
010100 _B : 18.0%	101110 _B : 28.4%	010100 _B : 30.0%	101110 _B : 40.4%		
010101 _B : 18.4%	101111 _B : 28.8%	010101 _B : 30.4%	101111 _B : 40.8%		
010110 _B : 18.8%	110000 _B : 29.2%	010110 _B : 30.8%	110000 _B : 41.2%		
010111 _B : 19.2%	110001 _B : 29.6%	010111 _B : 31.2%	110001 _B : 41.6%		
011000 _B : 19.6%	110010 _B : 30.0%	011000 _B : 31.6%	110010 _B : 42.0%		
011001 _B : 20.0%	All Others: Reserved	011001 _B : 32.0%	All Others: Reserved		

图 10 MR14 寄存器对照表

4. 在电子眼图使用中的常见问题

4.1. 打印串口的更改

我们 EVM 的串口默认是主域的 Uart0，因此电子眼图工具默认也是在 Uart0 下进行打印，但由于客户的板卡设计阶段各种因素的影响，客户可能不会将 UART0 的输出进行连接，可能会使用其他的串口进行输出，此

时就涉及到打印串口的更改。在电子眼图工具中，TI 提供了一个简便的方式对 UART 口进行更改。在 gel 文件夹下，Jacinto7_DDRSS_Tools_R5.gel 中通过注释 UART0 和打开 UART1 可以进行输出串口从 0 到 1 的更改，同时原 src 文件夹下对源代码中的 instance 进行更改，更改后需要重新编译，支持主域的 0 到 9 串口（更改到 0 和 1 以外的串口需要客户自行添加 pinmux 代码）。此时执行 UART IO Config 后会将对应输出打印在相应串口上，本文以 UART0 更改到 UART1 为例。

```

menuitem "J721E UART IO Config";

hotmenu J721E_Configure_UART0_PINMUX()
{
    Write_MMR(0x0011D008, 0x68EF3490);
    Write_MMR(0x0011D00C, 0xD172BC5A);

    // UART0
    Write_MMR(0x0011C1E8, 0x00050000); //RX
    Write_MMR(0x0011C1EC, 0x00010000); //TX
}

/*hotmenu J721E_Configure_UART1_PINMUX()
{
    Write_MMR(0x0011D008, 0x68EF3490);
    Write_MMR(0x0011D00C, 0xD172BC5A);

    // UART1
    Write_MMR(0x0011C1F8, 0x00050000);
    Write_MMR(0x0011C1FC, 0x00010000);
}*/

```

除以上更改源码之后，重新编译的方式进行串口更改，本文提供使用工具自带固件在 runtime 更改输出串口的的方式。在 gel 脚本中更改以上后，按照第二章介绍步骤进行到 load 默认文件夹下的固件，此时点击一次单步运行后再点击一次右下角的 stepover（图中标注）后，将 Core Register 中的 R0 寄存器值更改为 1（根据客户串口更改 0-9 的值）。再点击运行，可以在 UART1 看到串口输出。

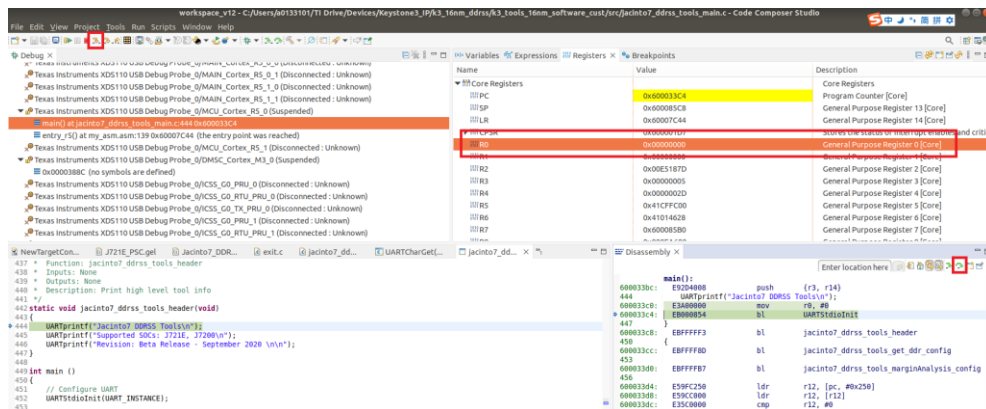


图 11 runtime 更改串口 instance 的示意图

4.2. Perl 脚本的运行

目前电子眼图工具的预处理脚本仅支持 Windows 命令行执行，在 Ubuntu 下执行会出现报错，无法运行。在 Windows 命令行下运行需要预先对 Perl 环境进行配置，在命令行中输入 perl -v，可以查看目前安装的 perl 的版本等信息，如无法运行则需要先安装 Perl 环境。本文安装的是 Perl 5, Version 30 的 Strawberry Perl，运行步骤如下图所示。注意运行时可能需要添加 Perl 的解释器，直接运行可能无输出。

```

C:\Users\A8501386\Downloads\jacinto7_ddrss_tools_beta\jacinto7_ddrss_tools_beta\perl\perl -v
This is perl 5, version 30, subversion 2 (v5.30.2) built for MSWin32-x64-multi-thread
Copyright 1987-2020, Larry Wall

Perl may be copied only under the terms of either the Artistic License or the
GNU General Public License, which may be found in the Perl 5 source kit.

Complete documentation for Perl, including FAQ lists, should be found on
this system using "man perl" or "perldoc perl".  If you have access to the
Internet, point your browser at http://www.perl.org/, the Perl Home Page.

C:\Users\A8501386\Downloads\jacinto7_ddrss_tools_beta\perl\perl.exe -x:\Strawberry\perl\bin\perl.exe jacinto7_ddrss_tools_marginAnalysis.pl LP4_Write_DlyStep16_VrefStep4_Example.txt
Processing input file LP4_Write_DlyStep16_VrefStep4_Example.txt and converting to csv format
Formatting the margin analysis output for dq bit 31
Formatting the margin analysis output for dq bit 30
Formatting the margin analysis output for dq bit 29
    
```

图 12 log 的处理与电子眼图的生成

5. 总结

电子眼图工具是 TI 为客户提供的对 DDR 的信号质量进行软件测量的工具，在实际的项目中，针对无法使用示波器在物理世界对眼图进行测量的情况下，提供一种在软件上对信号质量判断的依据，在客户量产项目中提供一种直观且有参考意义的调试工具。本文主要针对电子眼图的使用方法进行说明，并针对如何判断电子眼图结果的质量，以及电子眼图的工作原理进行了详细的阐述。

6. 参考文献

1. [TDA4VM Jacinto™ Processors for ADAS and Autonomous Vehicles Silicon Revisions 1.0 and 1.1 datasheet \(Rev. J\)](#)
2. [DRA829/TDA4VM/AM752x Technical Reference Manual \(Rev. B\)](#)
3. [Processor SDK RTOS J721E User Guide](#)
4. [Jacinto™ 7 DDRSS Register Configuration Tool User Guide](#)
5. [Jacinto™ 7 家族 HS 芯片中的 JTAG 加解锁控制](#)
6. [Jacinto 7 LPDDR4 Board Design and Layout Guidelines \(Rev. E\)](#)
7. Jacinto7 DDRSS Tools – User Guide

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