Freescale Semiconductor
Application Note White Paper

Rev. 4, 07/2015

i.MX6X DDR3 调校说明应用手册

by John Li(R64710)
GSM FAE
飞思卡尔半导体.
上海,

此应用手册主要说明如何调校 i.MX6X (包括 i.MX6Q/D/DL/S)的 DDR3.

	Date	Comment
V1	2013-09	• 创建文档 • 使用 DDR 压力测试工具，先前只测试 i.MX6Q SDP 板
V2	2014-09-19	• 根据测试工具最新版本 v1.0.3 更新文档 • 增加了新章节：应用 DDR3 校准参数到 Uboot；DRAM 接口高阶应用指导-DDR3 篇
V3	2015-03-26	• 3.3 节确认了不论是否 T-拓扑都需要做 write leveling 校准 • 修改了第 4 章，所有初始化脚本中的参数都需要在 uboot 中设置
V4	2015-07-30	• 插入第 4 章，支持各种 DDR3 配置，原有之后章节顺延

Contents

1 硬件资源，文档及工具下载

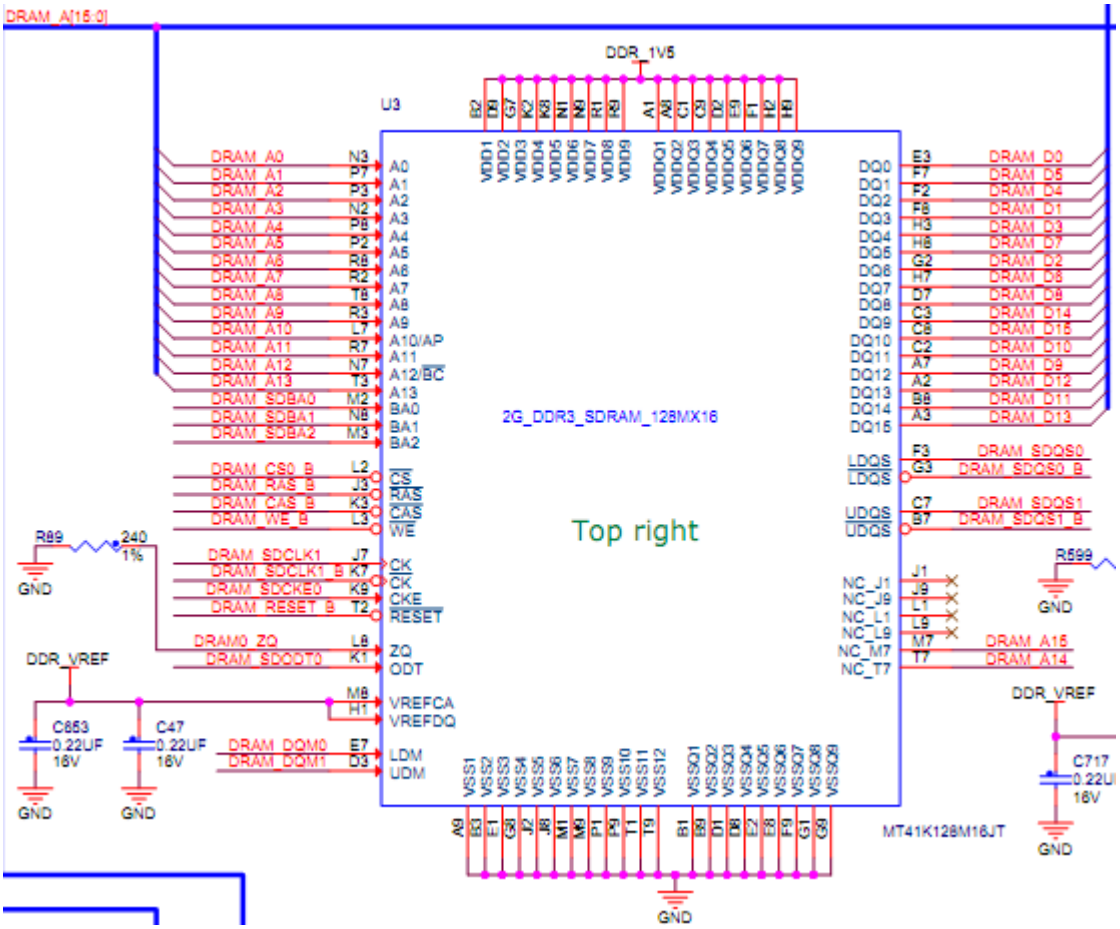
1.1 硬件资源

我们测试使用 i.MX6X SDP 板，使用 i.MX6Q 芯片，64bit 总线 DDR3(i.MX6D/DL 使用同一块 SDP 板)，硬件资源下载地址为:

http://www.freescale.com/zh-Hans/webapp/sps/site/prod_summary.jsp?code=i.MX6Q&fpsp=1&tab=Design_Tools_Tab

i.MX6_SABRE_SDP_DESIGNFILES : Design files, including hardware schematics, Gerbers, and OrCAD files..
大小 **(K):** 11631 格式**:** zip **Rev #:** C3 修改时间**:** 3/21/2013

SDP 板使用的 DDR3 为美光的：MT41K128M16JT



可以从美光网站下载其数据手册：

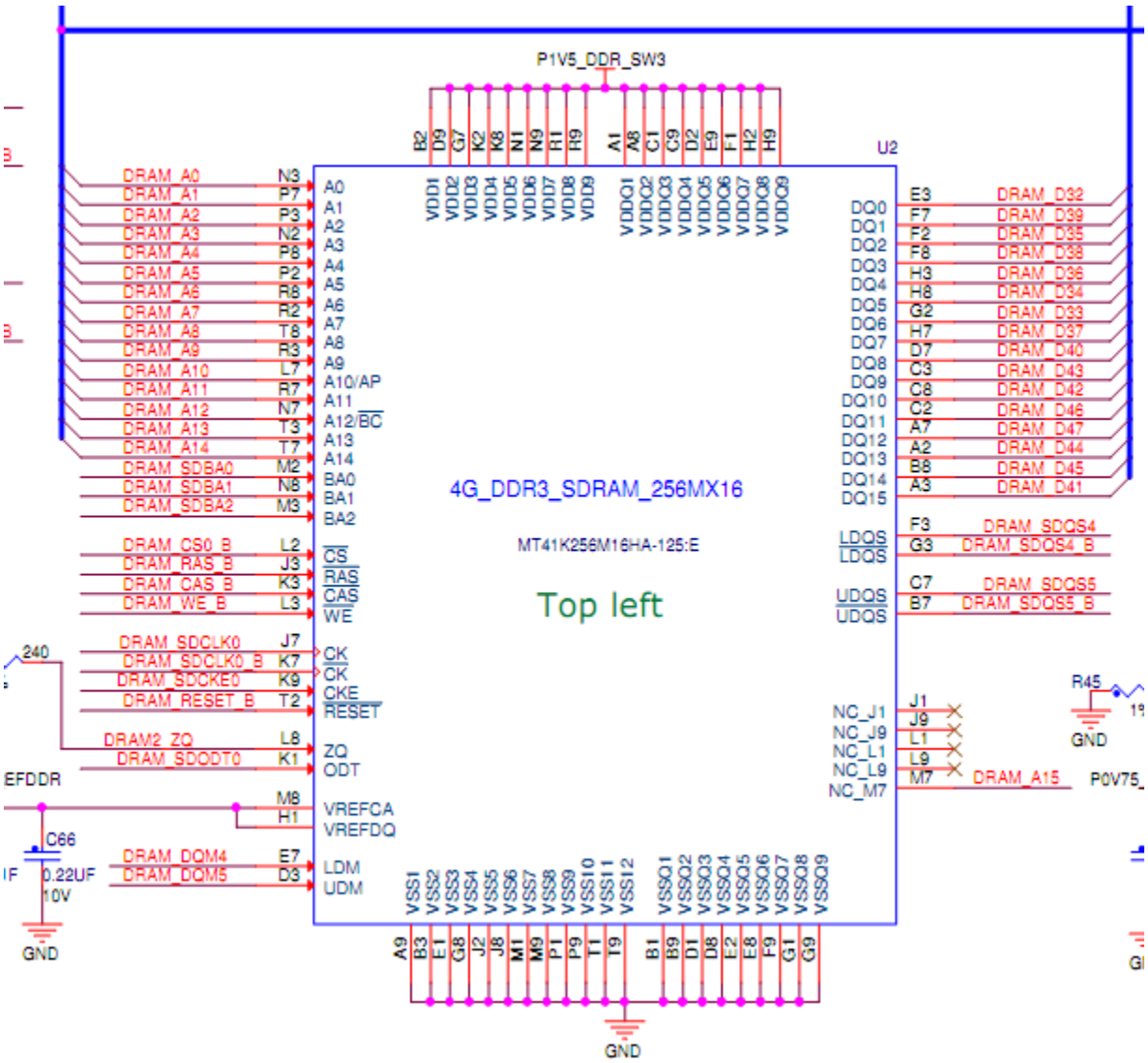
: http://www.micron.com/parts/dram/ddr3-sdram/mt41k128m16jt-107

我们使用 i.MX6 Saber AI 板，来做 i.MX6S，32bit 总线 DDR3 的测试，Saber AI 板的下载地址为:

http://www.freescale.com/zh-Hans/webapp/sps/site/prod_summary.jsp?code=i.MX6S&fpsp=1&tab=Design_Tools_Tab

i.MX6_SABRE_AI_DESIGNFILES : Design files, including hardware schematics, layout files and BOM.
大小 (K): 32886 格式: zip Rev #: 1.0.2 修改时间: 5/19/2014

它使用美光的: MT41K256M16HA-125:E, 如下图所示:



数据手册下载地址为: http://www.micron.com/parts/dram/ddr3-sdram/mt41k256m16ha-125.

注意从美光公开网站下载的数据手册可能缺少一些关键的参数，如时序和页大小之类的，所以最好从供应商获取最全的数据手册。

1.2 DDR3 调校相关的文档

除了 DDR3 的数据手册，我们可能也需要 JEDEC DDR3 规范，可以从以下地址下载，需要注册： www.jedec.org/sites/default/files/docs/JESD79-3D.pdf

除此之外，还需要参考 i.MX6X 的用户手册，和关于 DDR 校准的应用手册文档，下载地址如下：

http://www.freescale.com/zh-Hans/webapp/sps/site/prod_summary.jsp?code=i.MX6Q&fpsp=1&tab=Design_Tools_Tab

AN4467
i.MX 6 Series DDR Calibration

IMX6DQRM
i.MX 6Dual/6Quad Applications Processor Reference Manual

IMX6DQ6SDLSRM
Security Reference Manual for i.MX 6

1.3 DDR3 调校用的 DDR 压力测试工具.

飞思卡尔使用一个通过 USB(otg)连接的工具，来测试，校准目标板上的 DDR，名称为”DDR_Stress_Tester_V1.03”，下载地址为https://community.freescale.com/docs/DOC-96412

。

1.4 配置 DDR3 相关参数.

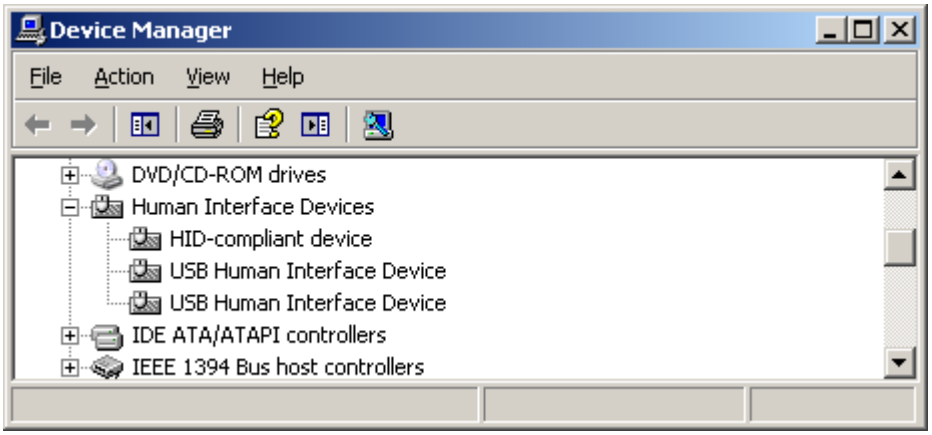
DDR3 相关参数配置比较复杂，飞思卡尔准备了一个 excel 工具来简化这个过程，可以从飞思卡尔社区连接下载https://community.freescale.com/docs/DOC-94917, 目前有 i.Mx6DQSDL DDR3 Script Aid, i.Mx6DQSDL LPDDR2 Script Aid 和 i.Mx6SL LPDDR2 Script Aid. 我们使用 i.Mx6DQSDL DDR3 Script Aid. 请在使用前参考它的 readme 页.

2 硬件连接

DDR3 测试工具要求工作在 i.MX6X 的下载模式下(BOOT_MODE[1:0]=0b01), (注意 i.MX6X 的 rom code 如果不能正常的从内部启动模式启动起来，它会自动进入下载模式，SDP 和 AI 板都使用这个机制来进入下载模式，而没有改动 BOOT_MODE 管脚), DDR3 测试工具会将测试使用的固件下载到 i.MX6X 的内部 RAM 中，用于测试外部的 DDR3。

如 SDP 板，设备　SW6 为(2,7 脚为 on,其它管脚为 off)，此为 SD3 启动模式，但如果不在 SD3 插槽上插入 SD 卡，然后启动，就会自动进入下载模式。

先使用 USB 线连接 PC 和 SDP 板，连接口为 USB OTG 口，J505。然后插入 5V 电源，电源接口为 P1。Rom code 在进入下载模式后，首先会通过 USB 口上报自己是一个 USB HID 设备，所以你在 PC 端的设备管理器中可以看到有 USB HID 设备插入，如下图所示:



这意味着 i.MX6X 板子已经进入下载模式，硬件连接如下图：

3 DDR3 调校步骤

DDR3 的调校包括以下几步：

1. 根据 DDR3 数据手册，JEDEC DDR3 规范，使用最新的 Mx6DQSDL DDR3 Script Aid V0.10.xlsx (https://community.freescale.com/docs/DOC-94917)工具，生成 DDR3 的初始化脚本。
2. 运行 DDR 压力测试工具来获得自动校准的值(不同的拓朴结构，需要校准的值可能不同)。
3. 将校准后的值填回到 DDR3 初始化脚本中，进行 DDR3 的压力测试。

3.1 生成 DDR3 初始化脚本

以 DDR3 MT41K128M16JT 为例，使用 DDR3 的脚本生成工具，我们需要输入其数据手册中的以下参数：

128 Meg x 16
16 Meg x 16 x 8 banks
8K
16K (A[13:0])
8 (BA[2:0])
1K (A[9:0])
2KB

- Memory type of -125 number is: DDR3-1600
- DRAM density(Gb)=128M x 16=2Gb
- DRAM Bus Width=16bit
- Number of Banks=8
- Number of ROW Addresses=14 A[13:0]
- Number of COLUMN=10 A[9:0]
- Page Size(K)=2

Table 1: Key Timing Parameters

Speed Grade	Data Rate (MT/s)	Target ^{t}RCD-^{t}RP-CL	^{t}RCD (ns)	^{t}RP (ns)	CL (ns)
-107[1, 2, 3]	1866	13-13-13	13.91	13.91	13.91
-125[1, 2]	1600	11-11-11	13.75	13.75	13.75
-15E[1]	1333	9-9-9	13.5	13.5	13.5
-187E	1066	7-7-7	13.1	13.1	13.1

Notes: 1. Backward compatible to 1066, CL = 7 (-187E).
2. Backward compatible to 1333, CL = 9 (-15E).
3. Backward compatible to 1600, CL = 11 (-107).

- tRCD=tRP=CL(ns)=13.75

Table 56: DDR3-1600 Speed Bins

DDR3-1600 Speed Bin		-125[1]			
CL-^{t}RCD-^{t}RP		11-11-11			
Parameter	Symbol	Min	Max	Unit	Notes
Internal READ command to first data	^{t}AA	13.75	–	ns	
ACTIVATE to internal READ or WRITE delay time	^{t}RCD	13.75	–	ns	
PRECHARGE command period	^{t}RP	13.75	–	ns	
ACTIVATE-to-ACTIVATE or REFRESH command period	^{t}RC	48.75	–	ns	
ACTIVATE-to-PRECHARGE command period	^{t}RAS	35	9 x ^{t}REFI	ns	2

- tRC Min(ns)=48.75
- tRAS Min(ns)=35

针对 i.MX6X 的 MMDC 配置如下:

- Bus width=64(i.MX6S 仅支持 32 bit, i.MX6Q/D/DL 可以支持 64 或 32 bit)
- Number of Chip Selects used=1(我们的 SDP 设计使用了一个 CS0 来连接 4 片 DDR3)
- DRAM Clock Freq(Mhz)=528(i.MX6Q/D 工作在 528Mhz, i.MX6DL/S 工作在 400Mhz)
- DRAM Clock Cycle Time(ns)=1.894(i.MX6Q/D=1/528Mhz=1.894ns, i.MX6DL/S=1/400Mhz=2.5ns)

注意 i.MX6X 的 MMDC 仅支持工作在 528Mhz 或 400Mhz，所以虽然我们选择的 DDR3 可以工作在 800Mhz(DDR3-1600)，所以我们在选择 DDR3 的时序参考时，我们使用 DDR3-1600 的时间参数(单位 nS) , 而不是时钟周期参数(clock),因为时钟周期是按照 800MHz 来计算的，不正确。

关于阻抗与端接设备，需要与 PCB 实际设计一致，我们默认使用:

- DRAM DSE Setting-DQ/DQM(ohm)=48
- DRAM DSE Setting-ADDR/CMD/CTL(ohm)=48
- DRAM DSE Setting-CK(ohm)=48
- DRAM DSE Setting-DQS(ohm)=48

- System ODT Setting(ohm)=60

在获得了我们所需的 DDR3 相关的所有参数后，我们就可以填写到 Mx6DQSDL DDR3 Script Aid V0.10.xlsx excel 表的 Register Configuration 页中，如下：

Device Information	
Manufacturer:	Micron
Memory part number:	MT41K128M16JT-125
Memory type:	DDR3-1600
DRAM density (Gb)	2
DRAM Bus Width	16
Number of Banks	8
Number of ROW Addresses	14
Number of COLUMN Addresses	10
Page Size (K)	2
Self-Refresh Temperature (SRT)	Normal
tRCD=tRP=CL (ns)	13.75
tRC Min (ns)	48.75
tRAS Min (ns)	35
System Information	
i.Mx Part	i.Mx6Q
Bus Width	64
Density per chip select (Gb)	8
Number of Chip Selects used	1
Total DRAM Density (Gb)	8
DRAM Clock Freq (MHz)	528
DRAM Clock Cycle Time (ns)	1.894
Address Mirror (for CS1)	Disable
SI Configuration	
DRAM DSE Setting - DQ/DQM (ohm)	48
DRAM DSE Setting - ADDR/CMD/CTL (ohm)	48
DRAM DSE Setting - CK (ohm)	48
DRAM DSE Setting - DQS (ohm)	48
System ODT Setting (ohm)	60

然后我们就可以从 Realview.inc 页中获得 DDR3 的初始化设置，拷贝出来保存在一下 txt 文件中，然后改名为：“MX6QD_SabreSD_DDR3_528MHz_64bit.inc”。

打开此脚本，修改以下项:

```
//==========================================================================

// Disable  WDOG
//==========================================================================
```

```
// setmem /16     0x020bc000 =      0x30   //注掉此项，DCD 不可以访问此地址
```

注掉以下的校准值，使用寄存器默认的值，如果第一次跑 DDR3 校准，也可以不注释掉，DDR3 校准时会自动更新相关校准值。

```
// For target board, may need to run write leveling calibration to fine tune these settings.

//setmem /32      0x021b080c  =     0x00000000
//setmem /32      0x021b0810 =      0x00000000
//setmem /32      0x021b480c  =     0x00000000
//setmem /32      0x021b4810 =      0x00000000

////Read DQS Gating calibration
//setmem /32      0x021b083c =      0x00000000  // MPDGCTRL0 PHY0
//setmem /32      0x021b0840 =      0x00000000  // MPDGCTRL1 PHY0
//setmem /32      0x021b483c =      0x00000000  // MPDGCTRL0 PHY1
//setmem /32      0x021b4840 =      0x00000000  // MPDGCTRL1 PHY1

//Read calibration
//setmem /32      0x021b0848 =      0x40404040  // MPRDDLCTL PHY0
//setmem /32      0x021b4848 =      0x40404040  // MPRDDLCTL PHY1

//Write calibration
//setmem /32      0x021b0850 =      0x40404040  // MPWRDLCTL PHY0
//setmem /32      0x021b4850 =      0x40404040  // MPWRDLCTL PHY1
```

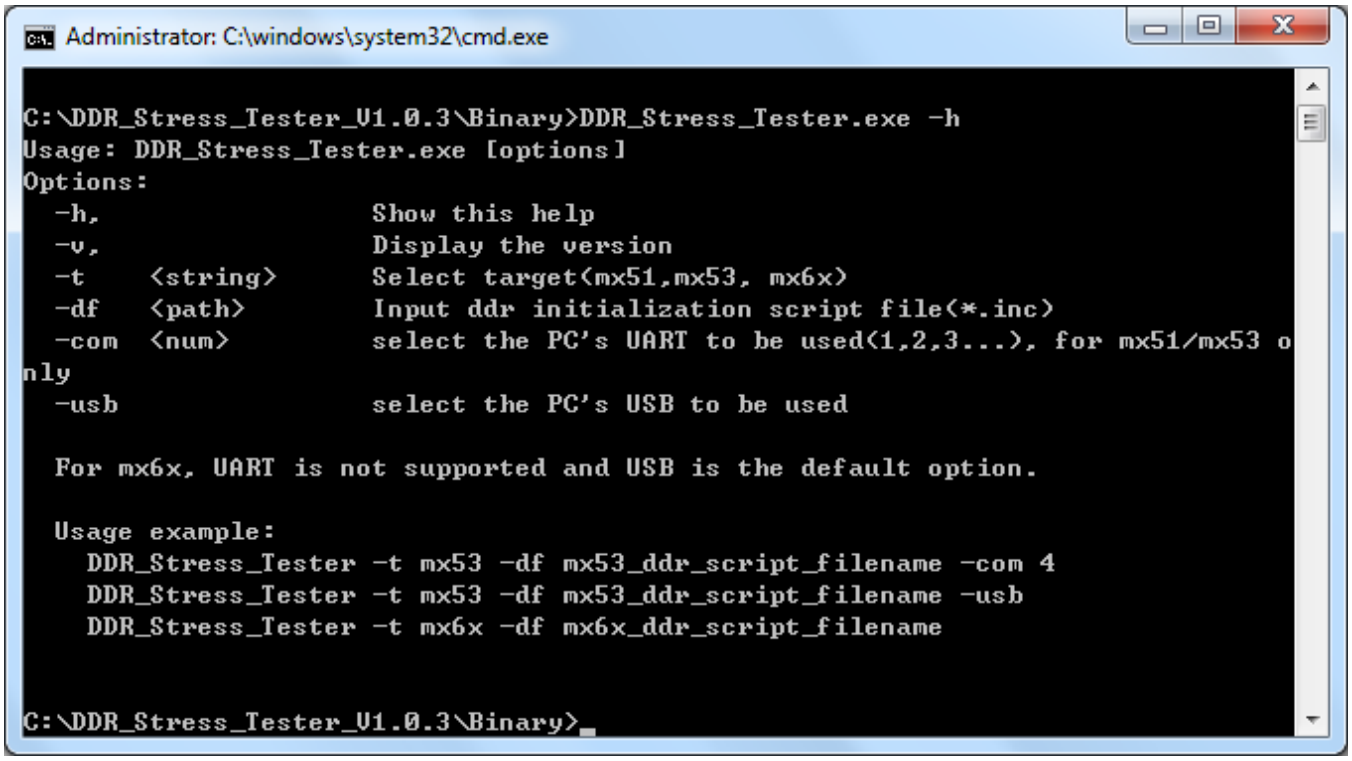
校准后，需要回填这些值。

3.2 使用 DDR3 压力测试工具校准 DDR3

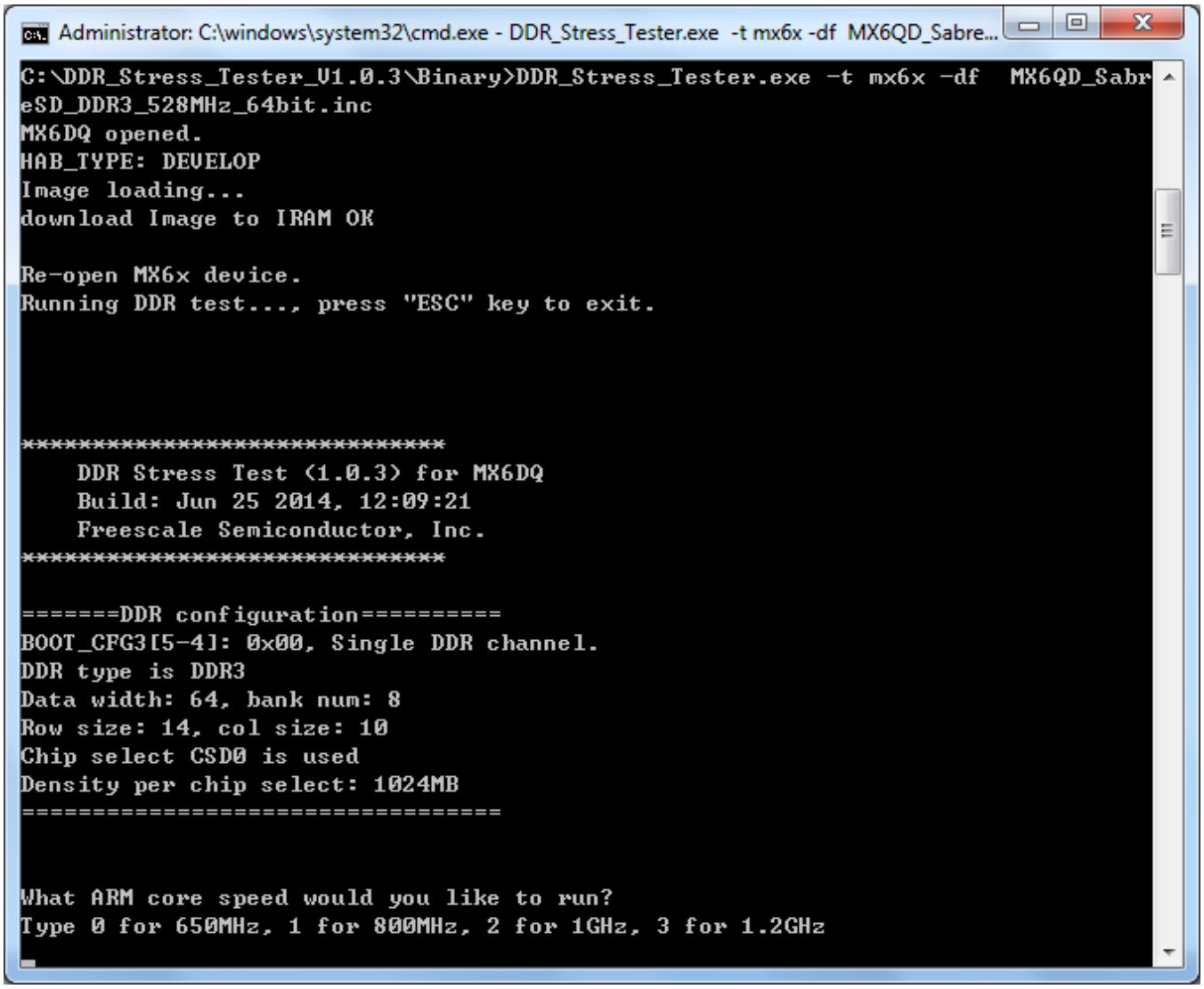
在 PC 上，使用 cmd 窗口，进入 DDR_Stress_Tester_Vxxx\binary 目录，此目录应该包括以下文件：

- DDR_Stress_Tester.exe: PC 端测试工具
- ddr-stress-test-mx6x.bin: 下载到目标板上的测试固件。
  MX6QD_SabreSD_DDR3_528MHz_64bit.inc: 刚刚使用工具生成的 DDR3 初始化脚本

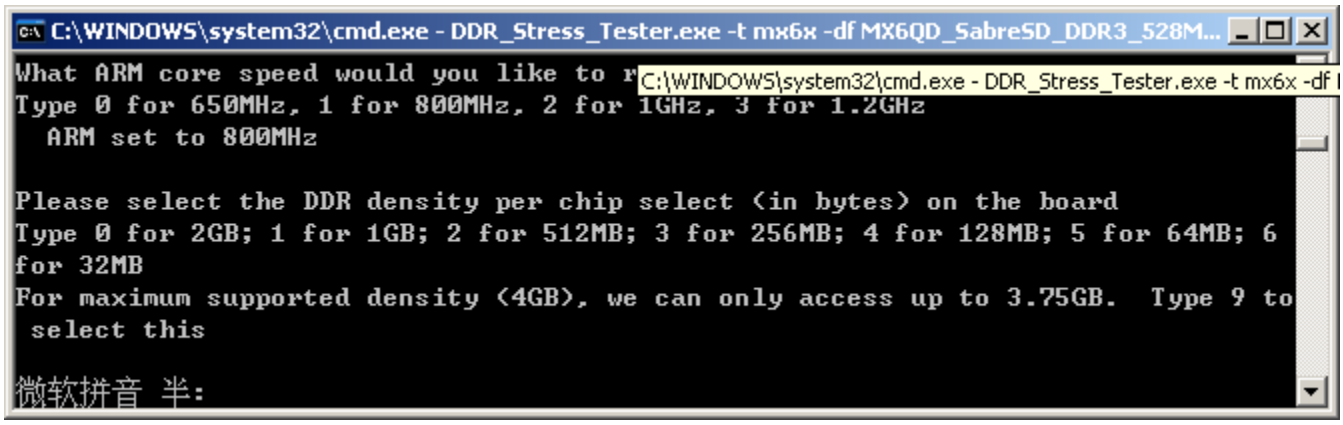
运行以下命令 “DDR_Stress_Tester.exe -h” 可以获得工具帮助:



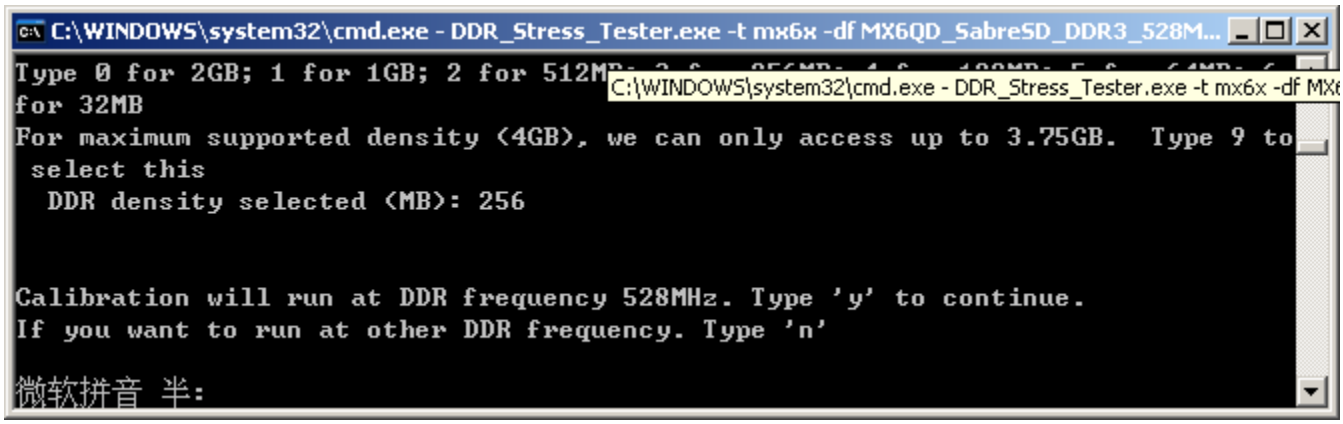
运行以下命令“DDR_Stress_Tester.exe -t mx6x -df MX6QD_SabreSD_DDR3_528MHz_64bit.inc” 得到以下信息:



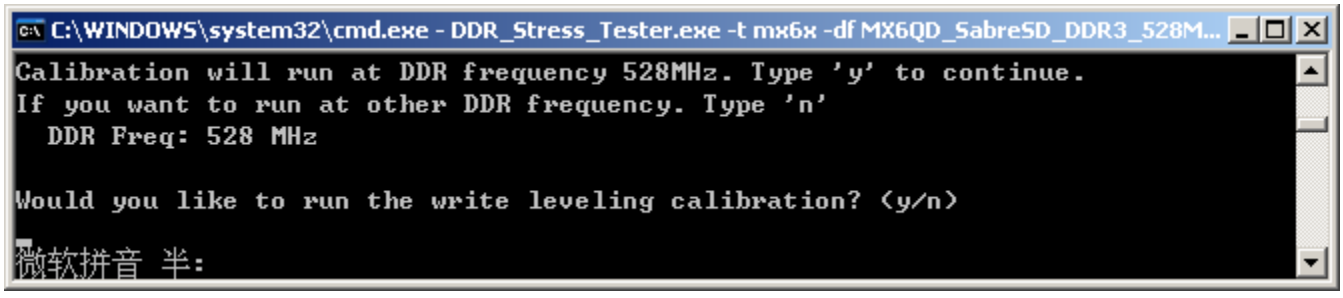
确认 DDR3 相关信息后，工具提示输入 ARM 工具频率，我们选择 800MHz,对于 1G 的芯片是中间频率。



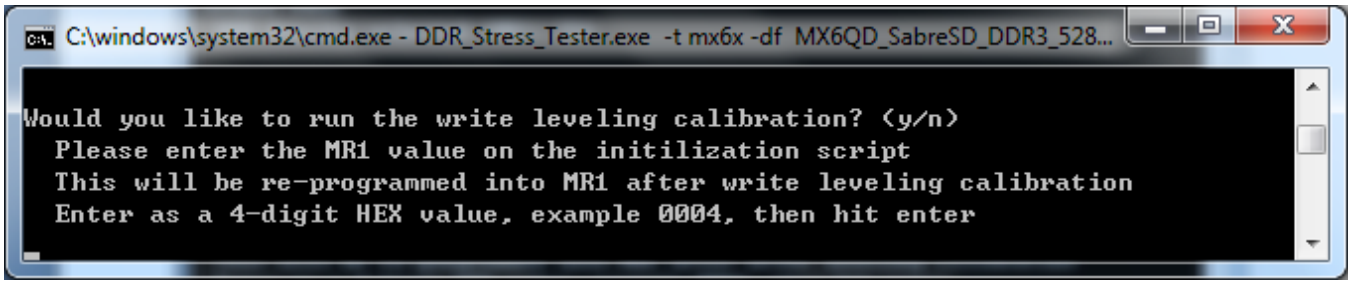
选择需要测试多大的 DDR3 空间，对于校准来说，选任何一下都可以，我们先选择 3 for 256MB



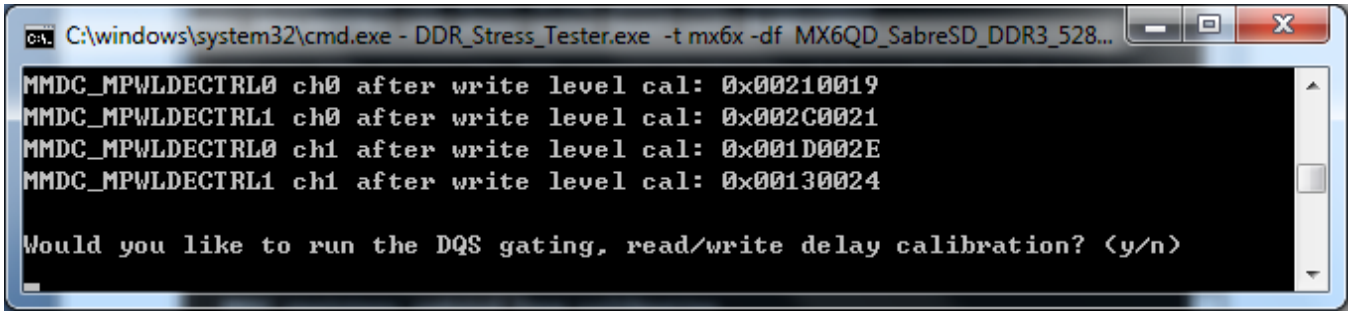
输入”y”表示使用 528Mhz 来进行校准。



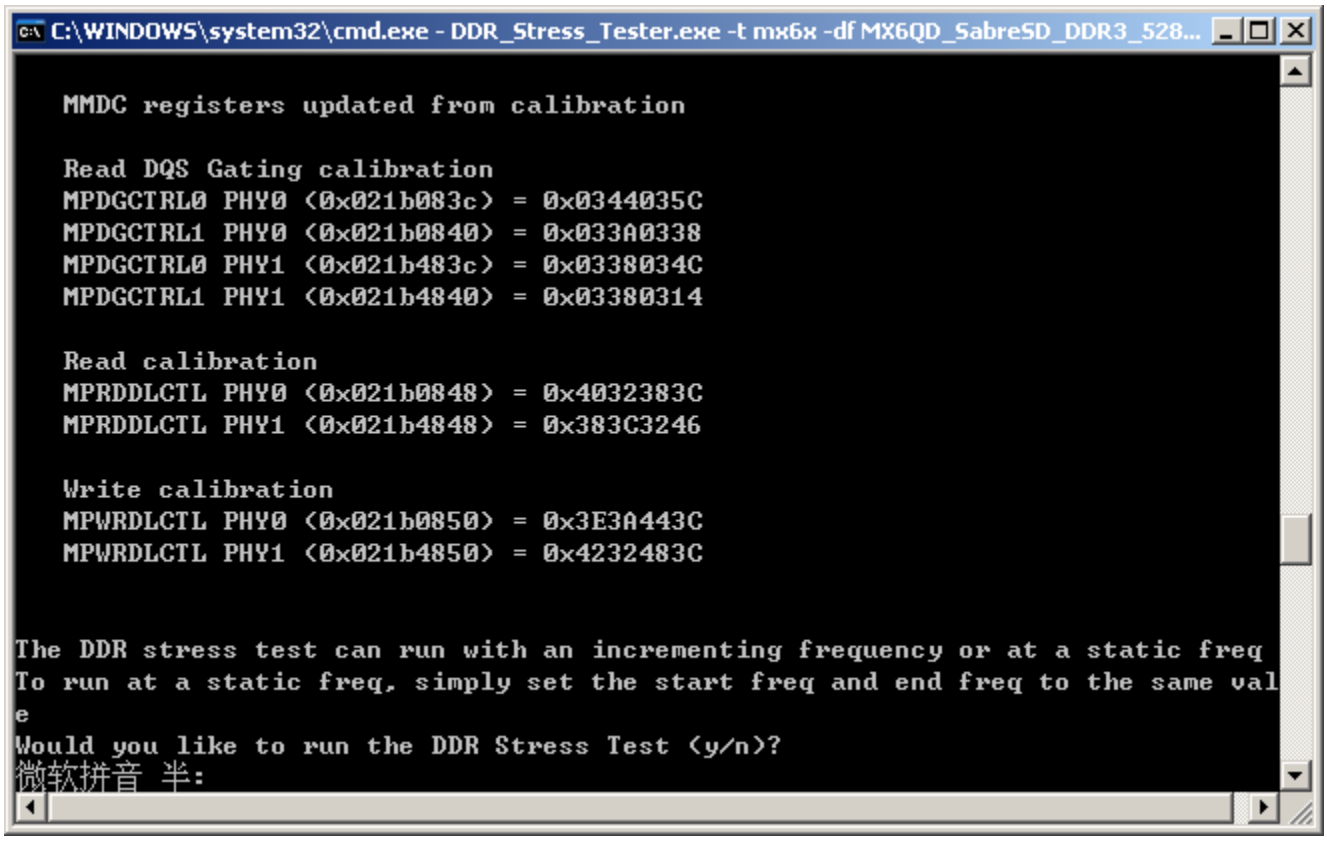
输入”y”表示需要进行 write leveling 校准。



输入 0004 以进行 write leveling 校准。



选择”y”来进行 DQS gating， read/write delay 校准(i.MX6X MMDC 硬件自动校准, 关闭 cache)。



校准结束后，可以直接进入压力测试，或是 ctrl-C 结束测试，将获得的校准值写回到初始化脚本中。

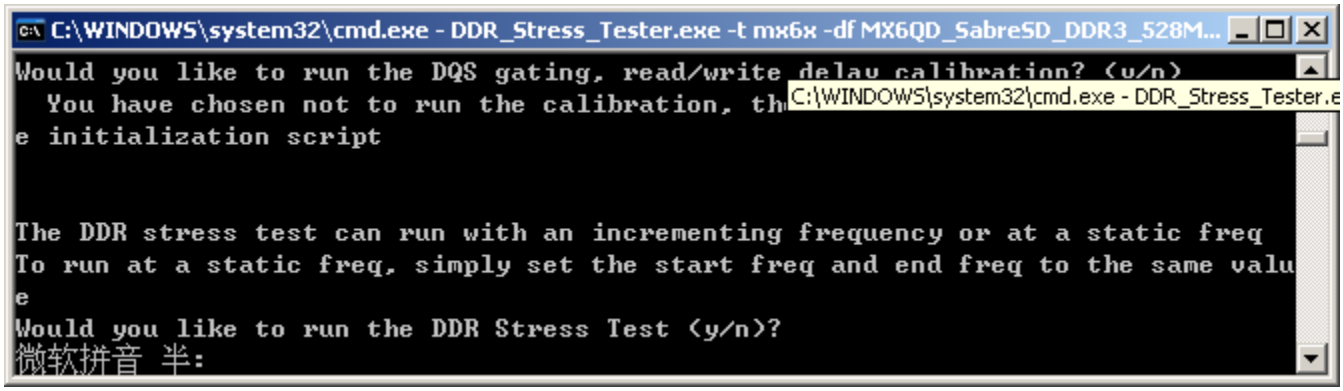
```
// For target board, may need to run write leveling calibration to fine tune these settings.

setmem /32  0x021b080c  =    0x001F001F
setmem /32  0x021b0810  =    0x002A0023
setmem /32  0x021b480c  =    0x001C002B
setmem /32  0x021b4810  =    0x000E0024

////Read DQS Gating calibration
setmem /32  0x021b083c  =     0x0344035c  // MPDGCTRL0 PHY0
setmem /32  0x021b0840  =     0x033a0338  // MPDGCTRL1 PHY0
setmem /32  0x021b483c  =     0x0338034c  // MPDGCTRL0 PHY1
setmem /32  0x021b4840  =     0x03380314  // MPDGCTRL1 PHY1

//Read calibration
setmem /32  0x021b0848  =     0x4032383c  // MPRDDLCTL PHY0
setmem /32  0x021b4848  =     0x383c3246  // MPRDDLCTL PHY1

//Write calibration
setmem /32  0x021b0850  =     0x3e3a443c  // MPWRDLCTL PHY0
setmem /32  0x021b4850  =     0x4232483c  // MPWRDLCTL PHY1
```

3.3 进行 DDR 压力测试.

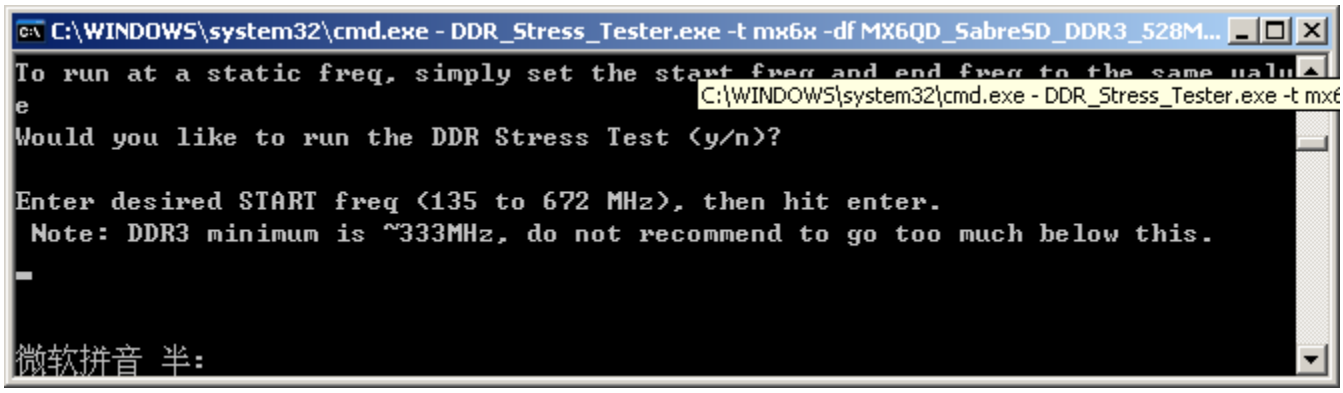
重启 i.MX6Q SDP 板, 再次运行命令“DDR_Stress_Tester.exe –t mx6x –df MX6QD_SabreSD_DDR3_528MHz_64bit.inc”, 选择:

- 1 for 800MHz ARM core
- 3 for 256MB DDR density
- y for run calibration with 528MHz
- n for not run the write leveling calibration
- n for not run the DQS gating, read/write delay calibration

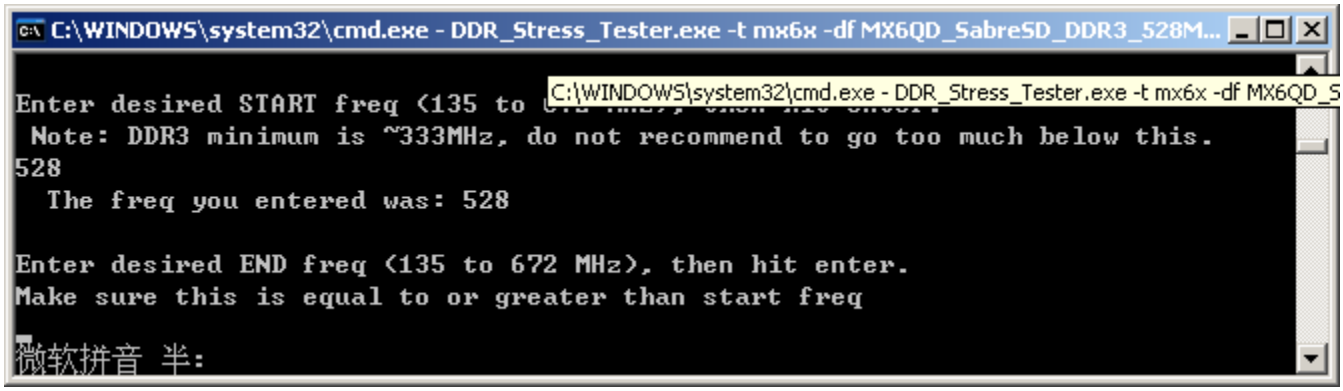
  然后系统提示是否进行压力测试?



- 选择 y 进行压力测试

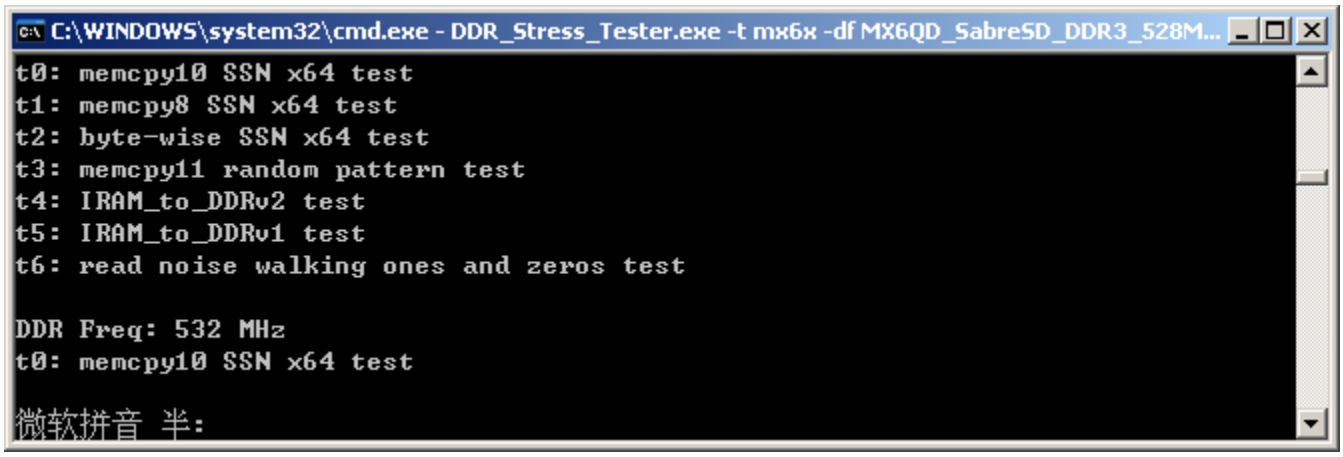


- 选择测试开始的最低频率，因为 i.MX6Q/D 运行在 528MHz,所以我们选择 528MHz 作为最低频率。

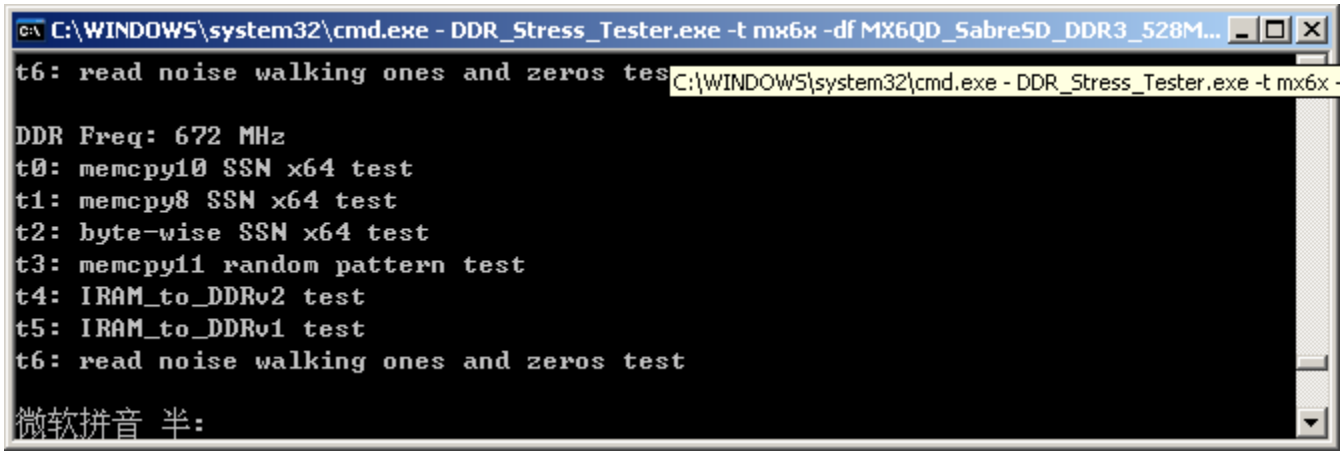


- 最高频率我们选可以支持的最高为 672Mhz。

然后工具会进行频率渐升的压力测试，每一个频率点进行 7 项测试。



一般我们认为如果可以工作在标准频率的 1.1 到 1.15 倍，则 DDR3 运行稳定，所以为 528x(1.1~1.15)=580Mhz~607Mhz，实测中可以看见 i.MX6X SDP 最高在常温下可以工作在 672MHz. 工作通过最高频率测试后又会从最低频率重新作升频测试。



然后，可以进行长时间测试：重新运行测试程序，不选校准，进入压力测试后，最低最高频率都设为 528MHz,就可以在此频率长时间运行，可以进行比如说一夜，12 小时测试，如果通过则认为 DDR3 稳定。

之后可以进行高低温老化测试，如果不通过，则在不通过的温度重新校准，测试，然后再做高，低，常温测试，保证设置的参数适合所以工作情况。

4 其它 DDR 配置

以 i.MX6Q/D 为例，（i.MX6DL 与之类似，只是 DDR clock 只支持 400Mhz，i.MX6S 只支持 400Mhz 的 DDR clock 和 32bit 总线），我们分别列出 64bit 2GB 内存，32bit 1GB 内存，

32bit 512MB 内存，以及 64bit 4GB 内存(2 个片选)，32bit 2GB 内存(2 个片选)等各种 DDR 配置搭配的设置。

4.1 64bit 2GB 内存

如果我们使用 64bit，4 片 16bit 总线，每片 256 Meg X 16=512MB 的 DDR，则需要修改的地方包括：

1. 每片 DRAM density(Gb)=512MB=4Gb
2. 一般来讲 256 Meg X 16 的内存 Number of Row Addresses=15
3. Bus Width=64bit
4. 从 i.MX6X MMDC 方面，一个片选会选中 16Gb=2GB

Device Information	
Manufacturer:	Micron
Memory part number:	???
Memory type:	DDR3-1600
DRAM density (Gb)	4
DRAM Bus Width	16
Number of Banks	8
Number of ROW Addresses	15
Number of COLUMN Addresses	10
Page Size (K)	2
Self-Refresh Temperature (SRT)	Normal
tRCD=tRP=CL (ns)	13.75
tRC Min (ns)	48.75
tRAS Min (ns)	35
System Information	
i.Mx Part	i.Mx6Q
Bus Width	64
Density per chip select (Gb)	16
Number of Chip Selects used	1
Total DRAM Density (Gb)	16
DRAM Clock Freq (MHz)	528
DRAM Clock Cycle Time (ns)	1.894
Address Mirror (for CS1)	Disable

4.2 32bit 1GB 内存

如果我们使用 32bit，2 片 16bit 总线,每片 256 Meg X 16=512MB 的 DDR，则需要修改的地方包括：

1. 每片 DRAM density(Gb)=512MB=4Gb
2. 一般来讲 256 Meg X 16 的内存 Number of Row Addresses=15
3. Bus width=32bit
4. 从 i.MX6X MMDC 方面，一个片选会选中 8Gb=1GB

Device Information	
Manufacturer:	Micron
Memory part number:	???
Memory type:	DDR3-1600
DRAM density (Gb)	4
DRAM Bus Width	16
Number of Banks	8
Number of ROW Addresses	15
Number of COLUMN Addresses	10
Page Size (K)	2
Self-Refresh Temperature (SRT)	Normal
tRCD=tRP=CL (ns)	13.75
tRC Min (ns)	48.75
tRAS Min (ns)	35
System Information	
i.Mx Part	i.Mx6Q
Bus Width	32
Density per chip select (Gb)	8
Number of Chip Selects used	1
Total DRAM Density (Gb)	8
DRAM Clock Freq (MHz)	528
DRAM Clock Cycle Time (ns)	1.894
Address Mirror (for CS1)	Disable

4.3 32bit 512MB 内存

如果我们使用 32bit，2 片 16bit 总线，每片 128 Meg X 16=256MB 的 DDR，则需要修改的地方包括：

1. 每片 DRAM density(Gb)=256MB=2Gb
2. 一般来讲 128 Meg X 16 的内存 Number of Row Addresses=14
3. Bus width=32bit
4. 从 i.MX6X MMDC 方面，一个片选会选中 4Gb=512MB

Device Information	
Manufacturer:	Micron
Memory part number:	???
Memory type:	DDR3-1600
DRAM density (Gb)	2
DRAM Bus Width	16
Number of Banks	8
Number of ROW Addresses	14
Number of COLUMN Addresses	10
Page Size (K)	2
Self-Refresh Temperature (SRT)	Normal
tRCD=tRP=CL (ns)	13.75
tRC Min (ns)	48.75
tRAS Min (ns)	35
System Information	
i.Mx Part	i.Mx6Q
Bus Width	32
Density per chip select (Gb)	4
Number of Chip Selects used	1
Total DRAM Density (Gb)	4
DRAM Clock Freq (MHz)	528
DRAM Clock Cycle Time (ns)	1.894
Address Mirror (for CS1)	Disable

4.4 64bit 4GB 内存

如果我们使用 64bit， 4 片 16bit 总线，每片 1GB，每片内部有两个晶片，两个片选，每个片选选中 4 片中的一个晶片，一个晶片的大小是 256 Meg X 16=512MB，总共 4GB。

或是使用 8 片 16bit 总线，每片 512MB，用两个片选，每个片选选中 4 片(2GB)，总共 4GB。

则需要修改的地方包括：

1. 每片 DRAM density(Gb)=512MB=4Gb
2. 一般来讲 256 Meg X 16 的内存 Number of Row Addresses=15

3. 从 i.MX6X MMDC 方面，一个片选会选中 16Gb=2GB.
4. Bus Width=64bit
5. 会使用两个片选，则 Total DRAM Density(Gb)=32Gb=4GB

Device Information	
Manufacturer:	Micron
Memory part number:	???
Memory type:	DDR3-1600
DRAM density (Gb)	4
DRAM Bus Width	16
Number of Banks	8
Number of ROW Addresses	15
Number of COLUMN Addresses	10
Page Size (K)	2
Self-Refresh Temperature (SRT)	Normal
tRCD=tRP=CL (ns)	13.75
tRC Min (ns)	48.75
tRAS Min (ns)	35
System Information	
i.Mx Part	i.Mx6Q
Bus Width	64
Density per chip select (Gb)	16
Number of Chip Selects used	2
Total DRAM Density (Gb)	32
DRAM Clock Freq (MHz)	528
DRAM Clock Cycle Time (ns)	1.894
Address Mirror (for CS1)	Disable

4.5 32bit 2GB 内存

如果我们使用 32bit， 2 片 16bit 总线，每片 1GB，内部有两个晶片，两个片选，每个片选选中 2 片中的一个晶片，一个晶片的大小是 256 Meg X 16=512MB，总共 2GB。

或是使用 4 片 16bit 总线，每片 512MB，用两个片选，每个片选选中 2 片(1GB)，总共 2GB。

则需要修改的地方包括：

1. 每片 DRAM density(Gb)=512MBX8=4Gb
2. 一般来讲 256 Meg X 16 的内存 Number of Row Addresses=15
3. 从 i.MX6X MMDC 方面，一个片选会选中 8Gb=1GB

4. Bus Width=64bit
5. 会使用两个片选，则 Total DRAM Density(Gb)=16Gb=2GB

Device Information	
Manufacturer:	Micron
Memory part number:	???
Memory type:	DDR3-1600
DRAM density (Gb)	4
DRAM Bus Width	16
Number of Banks	8
Number of ROW Addresses	15
Number of COLUMN Addresses	10
Page Size (K)	2
Self-Refresh Temperature (SRT)	Normal
tRCD=tRP=CL (ns)	13.75
tRC Min (ns)	48.75
tRAS Min (ns)	35
System Information	
i.Mx Part	i.Mx6Q
Bus Width	32
Density per chip select (Gb)	8
Number of Chip Selects used	2
Total DRAM Density (Gb)	16
DRAM Clock Freq (MHz)	528
DRAM Clock Cycle Time (ns)	1.894
Address Mirror (for CS1)	Disable

4.6 各种配置的区别

以上各种配置的区别如下表：

	64bit_1GB	64bit_2GB	64bit_4GB(2cs)	32bit_512MB	32bit_1GB	32bit_2GB(2cs)
MMDC0_MDCFG0 0x021b000c	0x54597955	0x898E7955	0x898E7955	0x54597955	0x898E7955	0x898E7955
MMDC0_MDOR 0x021b0030	0x00591023	0x008E1023	0x008E1023	0x00591023	0x008E1023	0x008E1023

Chan0 CS0_END 0x021b0040	0x00000027	0x00000047	0x00000047	0x00000017	0x00000027	0x00000027
MMDC0_MDCTL 0x021b0000	0x831A0000	0x841A0000	0xC41A0000	0x83190000	0x84190000	0xC4190000
MMDC0_MDSCR, MR2 write, CS0 0x021b001c	0x02088032	0x02088032	0x04088032	0x02088032	0x02088032	0x0408803A
MR0~MR3, ZQ cal write, CS1	N/A	N/A	0x0408803A 0x0000803B 0x00048039 0x19408038 0x04008048	N/A	N/A	0x0408803A 0x0000803B 0x00048039 0x19408038 0x04008048
DDR_PHY_P0_MPODTCTRL 0x021b0818	0x00022227	0x00022227	0x00011117	0x00022227	0x00022227	0x00011117
DDR_PHY_P1_MPODTCTRL 0x021b4818	0x00022227	0x00022227	0x00011117	N/A	N/A	N/A
DRAM_SDQS4~7 GRP_B4DS~B7DS DRAM_DQM4~7 Write leveling cal 2 high word Read DQS Gating cal 2high word				N/A	N/A	N/A

Read cal 1 high double word Write cal 1 high word P1_MPREDQBY0DL3~ P1_MPREDQBY3DL3 DDR_PHY_P1_MPMUR0, frc_msr DDR_PHY_P1_MPODTCTRL						

总结如下：

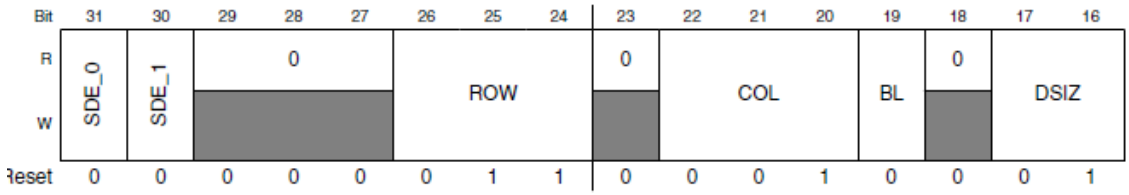
1. 尺寸相关:

- MMDC0_MDASP=0x021b0040, CS0_END

CS0_END	CS0_END. Defines the absolute last address associated with CS0 with increments of 256Mb. CS0_END=AXI_ADDRESS[31:25] bits. In DDR3 and 1-channel LPDDR2 mode: MMDCx_MDASP[CS0_END] should be set to DDR_CS_SIZE/32MB + 0x7 (DDR base address begins at 0x10000000) In 2-channel LPDDR2 mode: MMDC0_MDASP[CS0_END] should be set to DDR_CS_SIZE/32M + 0x3f (channel 0 base address begins at 0x80000000) MMDC1_MDASP[CS0_END] should be set to DDR_CS_SIZE/32M + 0x7 (channel 1 base address begins at 0x10000000) In 2-channel LPDDR2 with 4k-interleave mode: MMDC0_MDASP[CS0_END] should be set to DDR_CS_SIZE/32M + 0x43 MMDC1_MDASP[CS0_END] should be set to DDR_CS_SIZE/32M + 0x3

所以对一个片选选中 512MB，1GB，2GB。对应的值分别是 512/32+7=0x17，1024/32+7=0x27，2048/32+7=0x47。

- MMDC0_MDCTL=0x021b0000



Field	Description
31 SDE_0	MMDC Enable CS0. This bit enables/disables accesses from the MMDC toward Chip Select 0. The reset value of this bit is "0" (i.e No clocks and clock enable will be driven to the memory). At the enabling point the MMDC will perform an initialization process (including a delay on RESET and/or CKE) for both chip selects. The initialization length depends on the configured memory type.
	0 Disabled 1 Enabled
30 SDE_1	MMDC Enable CS1. This bit enables/disables accesses from the MMDC toward Chip Select 1. The reset value of this bit is "0" (i.e No clocks and clock enable will be driven to the memory). At the enabling point the MMDC will perform an initialization process (including a delay on RESET and/or CKE) for both chip selects. The initialization length depends on the configured memory type. 0 Disabled 1 Enabled
26–24 ROW	Row Address Width. This field specifies the number of row addresses used by the memory array. It will affect the way an incoming address will be decoded. Settings 110-111 are reserved 000 11 bits Row 001 12 bits Row 010 13 bits Row 011 14 bits Row 100 15 bits Row 101 16 bits Row
22–20 COL	Column Address Width. This field specifies the number of column addresses used by the memory array. It will determine how an incoming address will be decoded. 0x0 9 bits column 0x1 10 bits column 0x2 11 bits column 0x3 8 bits column 0x4 12 bits column 0x5-0xF Reserved
17–16 DSIZ	DDR data bus size. This field determines the size of the data bus of the DDR memory 0 16-bit data bus 1 32-bit data bus 2 64-bit data bus 3 Reserved —

所以对于一个片选，为 0X8xxxxxxx，对两个片选，为 0XCxxxxxxx。

对于 Row=14，为 0Xx3xxxxx，对于 Row=15，为 0Xx4xxxxxx。

对于 bus=64bit，为 0XxxxAxxxx，对于 bus=32bit，为 0Xxxx9xxxx。

2. 32bit/64bit，1cs/2cs 的区别（寄存器多寡）

- 对于 32bit，涉及到高 32bit 的相关比如说 pad 相关寄存器，calibration 相关寄存器，p1 相关寄存器可以不配。
- 对于 2cs，对 cs1 相关的 DDR MR0~MR3，ZQ calibration 寄存器需要再配一次。

3. Timing 相关

- 某些 timing 的参数与 ddr 尺寸相关，所以不同的尺寸有可能 timing 不同如：

MMDC0_MDCFG0=0x021b000c 的 tRFC, tXS

MMDC0_MDOR=0x021b0030 的 tXPR

4. 端接相关

- 对于 2cs 时，动态 ODT 的设置与 1cs 不同，如

MMDC0_MDSCR， MR2 write， CS0=0x021b001c 为:

M10	M9	Dynamic ODT ($R_{TT(WR)}$)
0	0	$R_{TT(WR)}$ disabled
0	1	RZQ/4 (60Ω [NOM])
1	0	RZQ/2 (120Ω [NOM])
1	1	Reserved

RZQ/4(60ohm Nom) for 1cs, RZQ/2(120ohm Nom) for 2cs.

DDR_PHY_P0_MPODTCTRL=0x021b0818 为：

10–8 ODT1_INT_RES	On chip ODT byte1 resistor - This field determines the Rtt_Nom of the on chip ODT byte1 resistor during read accesses. 0000 Rtt_Nom Disabled. 001 Rtt_Nom 120 Ohm 010 Rtt_Nom 60 Ohm 011 Rtt_Nom 40 Ohm 100 Rtt_Nom 30 Ohm 101 Rtt_Nom 24 Ohm 110 Rtt_Nom 20 Ohm 111 Rtt_Nom 17 Ohm

010 Rtt_nom=60ohm for 1cs, 001 Rtt_nom=120ohm for 2cs.

5. 软件配置时要注意两点：

- 对于 32bit 的配置，由于 DDR 配置参数减少，如果是在 3.0.x 内核中，DDR 的 DCD 段的总数和序号要相应调整，会比 64bit 要少。调整办法如下一章。
- 对于 2cs 的 DDR 尺寸大小的改动如下，以每个 cs 选中 1GB 为例:

Include\configs\Mx6q_sabresd.h

```
/*----------------------------------------------------------------------
 * Physical Memory Map
 */
#define CONFIG_NR_DRAM_BANKS            1
#define PHYS_SDRAM_1                    CSD0_DDR_BASE_ADDR
#define PHYS_SDRAM_1_SIZE               (1u * 1024 * 1024 * 1024)
#define iomem_valid_addr(addr, size) \
    (addr >= PHYS_SDRAM_1 && addr <= (PHYS_SDRAM_1 + PHYS_SDRAM_1_SIZE))
```

Change to

```
/*----------------------------------------------------------------------
 * Physical Memory Map
 */
#define CONFIG_NR_DRAM_BANKS            2
#define PHYS_SDRAM_1                    CSD0_DDR_BASE_ADDR
#define PHYS_SDRAM_1_SIZE               (1u * 1024 * 1024 * 1024)
#define PHYS_SDRAM_2                    CSD1_DDR_BASE_ADDR
#define PHYS_SDRAM_2_SIZE               (1u * 1024 * 1024 * 1024)
#define iomem_valid_addr(addr, size) \
    ((addr >= PHYS_SDRAM_1 && addr <= (PHYS_SDRAM_1 + PHYS_SDRAM_1_SIZE)) \
    || (addr >= PHYS_SDRAM_2 && addr <= (PHYS_SDRAM_2 + PHYS_SDRAM_2_SIZE)))
```

Board\freescale\mx6q_sabresd\Mx6q_sabresd.c

```
int dram_init(void)
{
    /*
     * Switch PL301_FAST2 to DDR Dual-channel mapping
     * however this block the boot up, temperory redraw
     */
    /*
     * u32 reg = 1;
     * writel(reg, GPV0_BASE_ADDR);
```

```
     */
    gd->bd->bi_dram[0].start = PHYS_SDRAM_1;
    gd->bd->bi_dram[0].size = PHYS_SDRAM_1_SIZE;
    return 0;
}
```

Change to

```
int dram_init(void)
{
    /*
     * Switch PL301_FAST2 to DDR Dual-channel mapping
     * however this block the boot up, temperory redraw
     */
    /*
     * u32 reg = 1;
     * writel(reg, GPV0_BASE_ADDR);
     */
    gd->bd->bi_dram[0].start = PHYS_SDRAM_1;
    gd->bd->bi_dram[0].size = PHYS_SDRAM_1_SIZE;
    gd->bd->bi_dram[1].start = PHYS_SDRAM_2;
    gd->bd->bi_dram[1].size = PHYS_SDRAM_2_SIZE;
    return 0;
}
```

5 DDR3 参数应用到 Uboot

为了使 DDR3 工作在较好的状态，应该在 uboot 中应用经过校准的参数。

经过校准的 DDR3 参数包括两个部分：

第一个部分：

由 script aid 生成的初始化脚本中的静态参数，具体参见 **3.1 生成 DDR3 初始化脚本**

第二个部分：

由 DDR3 压力测试工具校准后的参数，具体参见 **3.2 使用 DDR3 压力测试工具校准 DDR3**

Uboot 应该根据经过上述步骤得到的所有寄存器参数值，来逐一应用 DDR3 相关参数。

对于不同版本的 uboot，应用 DDR3 参数的方法稍有不同。

- L3.0.35.410 uboot 使用 DDR3 参数方法

该版本需要修改源代码 u-boot-2009.08/Board/freescale/${board_type}/flash_header.S，如下所示。

请注意源代码中的两个宏开关 CONFIG_MX6DL_DDR3 和 CONFIG_DDR_32BIT，找到正确的代码部分，然后根据上面得到的初始化脚本，设置该部分的所有寄存器。下面只列出了其中的一部分。

```
dcd_hdr:        .word 0x40E001D2 /* Tag=0xD2, Len=59*8 + 4 + 4, Ver=0x40 */
write_dcd_cmd:  .word 0x04DC01CC /* Tag=0xCC, Len=59*8 + 4, Param=0x04 */

MXC_DCD_ITEM(1, IOMUXC_BASE_ADDR + 0x774, 0x000C0000)
MXC_DCD_ITEM(2, IOMUXC_BASE_ADDR + 0x754, 0x00000000)

MXC_DCD_ITEM(3, IOMUXC_BASE_ADDR + 0x4ac, 0x00000028)
MXC_DCD_ITEM(4, IOMUXC_BASE_ADDR + 0x4b0, 0x00000028)
……..
//writing leveling calibration
MXC_DCD_ITEM(40, MMDC_P0_BASE_ADDR + 0x80c, 0x001F001F)
MXC_DCD_ITEM(41, MMDC_P0_BASE_ADDR + 0x810, 0x001F001F)
```

MXC_DCD_ITEM(42, MMDC_P1_BASE_ADDR + 0x80c, 0x001F001F)

MXC_DCD_ITEM(43, MMDC_P1_BASE_ADDR + 0x810, 0x001F001F)

//Read DQS gating calibration

MXC_DCD_ITEM(44, MMDC_P0_BASE_ADDR + 0x83c, 0x4333033F)

MXC_DCD_ITEM(45, MMDC_P0_BASE_ADDR + 0x840, 0x032C031D)

MXC_DCD_ITEM(46, MMDC_P1_BASE_ADDR + 0x83c, 0x43200332)

MXC_DCD_ITEM(47, MMDC_P1_BASE_ADDR + 0x840, 0x031A026A)

//Read calibration

MXC_DCD_ITEM(48, MMDC_P0_BASE_ADDR + 0x848, 0x4D464746)

MXC_DCD_ITEM(49, MMDC_P1_BASE_ADDR + 0x848, 0x47453F4D)

//Write calibration

MXC_DCD_ITEM(50, MMDC_P0_BASE_ADDR + 0x850, 0x3E434440)

MXC_DCD_ITEM(51, MMDC_P1_BASE_ADDR + 0x850, 0x47384839)

…..

关于 DCD 长度的修改

dcd_hdr: .word 0x40E001D2 /* Tag=0xD2, Len=59*8 + 4 + 4, Ver=0x40 */

write_dcd_cmd: .word 0x04DC01CC /* Tag=0xCC, Len=59*8 + 4, Param=0x04 */

dcd_hdr: 59*8 + 4 + 4 = 0x01E0 大小端关系交换字节序为 E001

write_dcd_cmd: 59*8 + 4 = 0x01DC 大小端关系交换字节序为 DC01

举例：

原 DCD 为

dcd_hdr: .word 0x40E001D2 /* Tag=0xD2, Len=59*8 + 4 + 4, Ver=0x40 */

write_dcd_cmd: .word 0x04DC01CC /* Tag=0xCC, Len=59*8 + 4, Param=0x04 */

MXC_DCD_ITEM(1, IOMUXC_BASE_ADDR + 0x774, 0x000C0000)

MXC_DCD_ITEM(2, IOMUXC_BASE_ADDR + 0x754, 0x00000000)

MXC_DCD_ITEM(3, IOMUXC_BASE_ADDR + 0x4ac, 0x00000030)
MXC_DCD_ITEM(4, IOMUXC_BASE_ADDR + 0x4b0, 0x00000030)

增加一行(红色部分)如下，需要修改 dcd_hdr, write_dcd_cmd 以及
重新调整 DCD command 序列号(红色部分)

dcd_hdr: .word 0x40E801D2 /* Tag=0xD2, Len=60*8 + 4 + 4, Ver=0x40 */
write_dcd_cmd: .word 0x04E401CC /* Tag=0xCC, Len=60*8 + 4, Param=0x04 */

MXC_DCD_ITEM(1, IOMUXC_BASE_ADDR + 0x774, 0x000C0000)
MXC_DCD_ITEM(2, IOMUXC_BASE_ADDR + 0x754, 0x00000000)
MXC_DCD_ITEM(3, IOMUXC_BASE_ADDR + 0x755, 0x00000000)

MXC_DCD_ITEM(4, IOMUXC_BASE_ADDR + 0x4ac, 0x00000030)
MXC_DCD_ITEM(5, IOMUXC_BASE_ADDR + 0x4b0, 0x00000030)

- L3.10.17_1.0.0/ L3.10.53_1.1.0 uboot 使用 DDR3 参数方法

该版本需要修改源代码 uboot-imx /board/freescale/${board_type}/mx6q_4x_mt41j128.cfg, 需修改部分如下所示:

```
/*
 * Device Configuration Data (DCD)
 *
 * Each entry must have the format:
 * Addr-type          Address        Value
 *
 * where:
 *   Addr-type register length (1,2 or 4 bytes)
 *   Address           absolute address of the register
 *   value     value to be stored in the register
 */
DATA 4, 0x020e0798, 0x000C0000
DATA 4, 0x020e0758, 0x00000000
DATA 4, 0x020e0588, 0x00000030
DATA 4, 0x020e0594, 0x00000030
DATA 4, 0x020e056c, 0x00000030
DATA 4, 0x020e0578, 0x00000030
DATA 4, 0x020e074c, 0x00000030
DATA 4, 0x020e057c, 0x00000030
DATA 4, 0x020e058c, 0x00000000
DATA 4, 0x020e059c, 0x00000030
DATA 4, 0x020e05a0, 0x00000030
DATA 4, 0x020e078c, 0x00000030
DATA 4, 0x020e0750, 0x00020000
DATA 4, 0x020e05a8, 0x00000028
DATA 4, 0x020e05b0, 0x00000028
DATA 4, 0x020e0524, 0x00000028
DATA 4, 0x020e051c, 0x00000028
```

```
DATA 4, 0x020e0518, 0x00000028
DATA 4, 0x020e050c, 0x00000028
DATA 4, 0x020e05b8, 0x00000028
DATA 4, 0x020e05c0, 0x00000028
DATA 4, 0x020e0774, 0x00020000
DATA 4, 0x020e0784, 0x00000028
DATA 4, 0x020e0788, 0x00000028
DATA 4, 0x020e0794, 0x00000028
DATA 4, 0x020e079c, 0x00000028
DATA 4, 0x020e07a0, 0x00000028
DATA 4, 0x020e07a4, 0x00000028
DATA 4, 0x020e07a8, 0x00000028
DATA 4, 0x020e0748, 0x00000028
DATA 4, 0x020e05ac, 0x00000028
DATA 4, 0x020e05b4, 0x00000028
DATA 4, 0x020e0528, 0x00000028
DATA 4, 0x020e0520, 0x00000028
DATA 4, 0x020e0514, 0x00000028
DATA 4, 0x020e0510, 0x00000028
DATA 4, 0x020e05bc, 0x00000028
DATA 4, 0x020e05c4, 0x00000028
DATA 4, 0x021b0800, 0xa1390003
DATA 4, 0x021b080c, 0x001F001F    // writing leveling calibration
DATA 4, 0x021b0810, 0x001F001F
DATA 4, 0x021b480c, 0x001F001F
DATA 4, 0x021b4810, 0x001F001F
DATA 4, 0x021b083c, 0x43200330    //Read DQS gating calibration
DATA 4, 0x021b0840, 0x031C0314
DATA 4, 0x021b483c, 0x4320032C
DATA 4, 0x021b4840, 0x031C0260
DATA 4, 0x021b0848, 0x3A2A3230    //Read calibration
```

```
DATA 4, 0x021b4848, 0x3E38343E
DATA 4, 0x021b0850, 0x36383E38    //Write calibration
DATA 4, 0x021b4850, 0x3C263E3E
DATA 4, 0x021b081c, 0x33333333
DATA 4, 0x021b0820, 0x33333333
DATA 4, 0x021b0824, 0x33333333
DATA 4, 0x021b0828, 0x33333333
DATA 4, 0x021b481c, 0x33333333
DATA 4, 0x021b4820, 0x33333333
DATA 4, 0x021b4824, 0x33333333
DATA 4, 0x021b4828, 0x33333333
DATA 4, 0x021b08b8, 0x00000800
DATA 4, 0x021b48b8, 0x00000800
DATA 4, 0x021b0004, 0x00020036
DATA 4, 0x021b0008, 0x09444040
DATA 4, 0x021b000c, 0x8A8F7955
DATA 4, 0x021b0010, 0xFF328F64
DATA 4, 0x021b0014, 0x01FF00DB
DATA 4, 0x021b0018, 0x00001740
DATA 4, 0x021b001c, 0x00008000
DATA 4, 0x021b002c, 0x000026d2
DATA 4, 0x021b0030, 0x008F1023
DATA 4, 0x021b0040, 0x00000047
DATA 4, 0x021b0000, 0x841A0000
DATA 4, 0x021b001c, 0x04088032
DATA 4, 0x021b001c, 0x00008033
DATA 4, 0x021b001c, 0x00048031
DATA 4, 0x021b001c, 0x09408030
DATA 4, 0x021b001c, 0x04008040
DATA 4, 0x021b0020, 0x00005800
DATA 4, 0x021b0818, 0x00011117
```

DATA 4, 0x021b4818, 0x00011117

DATA 4, 0x021b0004, 0x00025576

DATA 4, 0x021b0404, 0x00011006

DATA 4, 0x021b001c, 0x00000000

6 Appendix A: DDR3 基本知识

- DDR 内存

DDR=Double Data Rate 双倍速率同步动态随机存储器。严格的说 DDR 应该叫 DDR SDRAM，人们习惯称为 DDR，其中，SDRAM 是 Synchronous Dynamic Random Access Memory 的缩写，即同步动态随机存取存储器。而 DDR SDRAM 是 Double Data Rate SDRAM 的缩写，是双倍速率同步动态随机存储器的意思。它能够在时钟的上升期和下降期各传输一次数据, DDR 内存的频率可以用工作频率和等效频率两种方式表示，工作频率是内存颗粒实际的工作频率，但是由于 DDR 内存可以在脉冲的上升和下降沿都传输数据，因此传输数据的等效频率是工作频率的两倍.

- DDR3 与 DDR2 几个主要的不同之处：

1. 突发长度（Burst Length，BL):由于 DDR3 的预取为 8bit，所以突发传输周期（Burst Length，BL）也固定为 8
2. 寻址时序（Timing）:就像 DDR2 从 DDR 转变而来后延迟周期数增加一样，DDR3 的 CL 周期也将比 DDR2 有所提高。DDR2 的 CL 范围一般在 2～5 之间，而 DDR3 则在 5～11 之间，且附加延迟（AL）的设计也有所变化。DDR2 时 AL 的范围是 0～4，而 DDR3 时 AL 有三种选项，分别是 0、CL-1 和 CL-2。另外，DDR3 还新增加了一个时序参数——写入延迟（CWD），这一参数将根据具体的工作频率而定
3. DDR3 新增的重置（Reset）功能:重置是 DDR3 新增的一项重要功能，并为此专门准备了一个引脚
4. DDR3 新增 ZQ 校准功能:ZQ 也是一个新增的脚，在这个引脚上接有一个 240欧姆的低公差参考电阻。这个引脚通过一个命令集，通过片上校准引擎（On-Die Calibration Engine，ODCE）来自动校验数据输出驱动器导通电阻与 ODT 的终结电阻值。当系统发出这一指令后，将用相应的时钟周期（在加电与初始化之后用 512 个时钟周期，在退出自刷新操作后用 256 个时钟周期、在其他情况下用 64 个时钟周期）对导通电阻和 ODT 电阻进行重新校准
5. 参考电压分成两个
6. 点对点连接（Point-to-Point，P2P）

7 Appendix B: Write Leveling 设置

1. DDR layout 要求 DQS 信号长度要短于 CK 信号，但是 T-Topology 和 fly-by 不同，如下图。T-Topology 的每片 DDR 的 CK 信号与 DQS 长度差距相近，而 fly-by 长度差距不同.
2. 所以我们T-Topology,都使用相近的值0x1F,这个经验值。而fly-by的比较近的两片使用0x1F,比较远的两片使用 0x44。
3. T-Topology 就不需要做 write leveling calibration.而 fly-by 则需要。

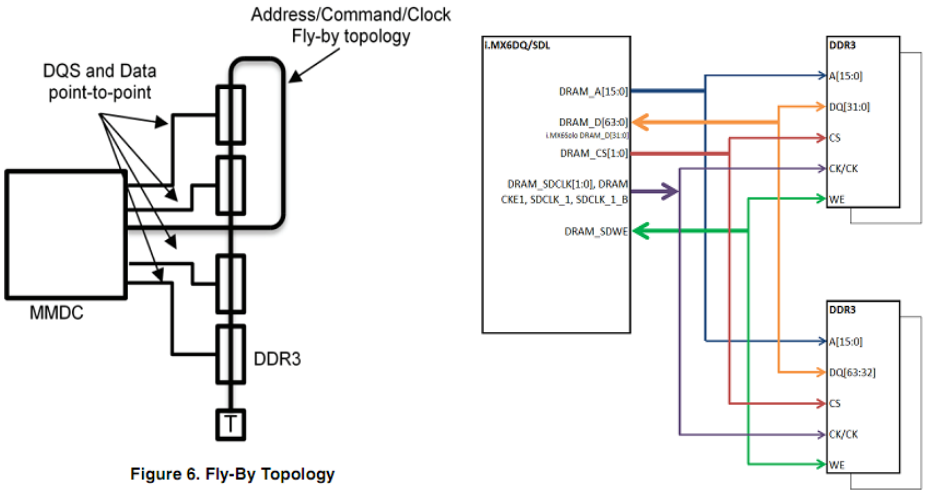


Figure 6. Fly-By Topology

Table 2-3. DDR3 routing by byte group

Chip signals	Group	Length		Recommendations
		Min	Max	
DRAM_D[7:0] DRAM_DQM0 DRAM_SDQS0 DRAM_SDQS0_B	Byte Group 1	—	Clock (min)	Match the signals of each byte group ± 25 mils. Match the differential signals of DQS ± 10 mils.

8 Appendix C: Write/read delay line 概念

- 读 DQS delay line calibration 用于在读 BYTE 中调节读 DQS 信号的 delay,当读 BYTE 数据从 DDR 发出时，相应 DQS 信号是与 BYTE 信号对齐的，当 MMDC 接收信号时，需要将 DQS 信号 delay 约 1/4 周期来对齐采样点，但是因为信号斜率的影响，需要手动调节对齐点，理论可以有从 0 到 1/2 周期的窗口。

- 写 DQS delay line calibration 与读类似，只是 delay 的是数据线

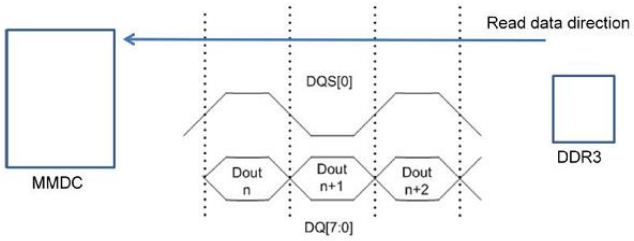


Figure 10. Data Read Illustration Showing Edge Aligned DQS with Data

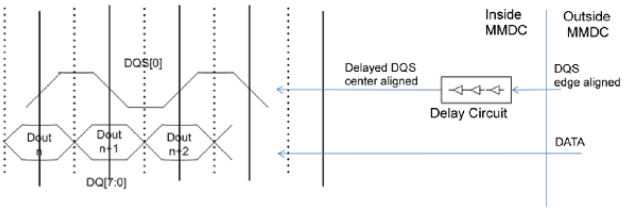


Figure 11. Illustration of DQS Being Delayed to Center Align it with Incoming Read Data

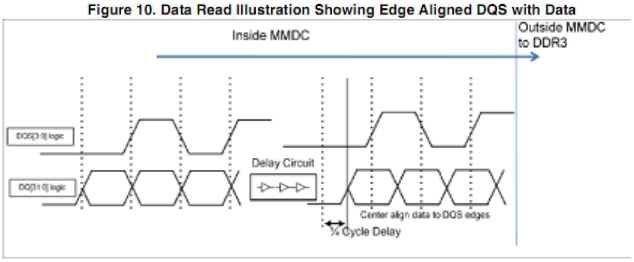


Figure 12. Illustration of Data Being Delayed to Center Align it with DQS

9 Appendix D: DRAM 接口高阶应用指导-DDR3 篇

https://community.freescale.com/docs/DOC-101694

基于如下原因，建议客户在 T-top 时候也做 write leveling calibration：

1. 数据线和地址线的长度相差太大；
2. DDR 芯片内部也有 delay 。

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
+1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 010 5879 8000
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature
Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Rev. 18
09/2013