说明:

在阅读本文档 SPI boot 原理之前，需要知道以下几点：

1. 选择 SPI boot 需要将拨码开关拨到特定的配置。（第一章节会描述）
2. 烧写工具是 TI 自带的 nor-writer，被烧写文件是由.out 经过一系列工具链转化后形成的.dat 镜像文件。（工具链的使用已经被简化成 bat 批处理文件，容易使用）
3. 镜像文件的逻辑顺序为：**boot parameter table**（256 字节，前 32 字节有用），**ddr configuration table**（可以没有，没有的话就不能再 boot 主代码前初始化 ddr），**核 0 的数据部分，核 1 的数据部分....**（每个核的数据部分又可以细分，可参见多核BOOT **研究**）
4. 知道 boot 的流程是先编写合适的工程（多核存储空间不重叠，核 0 具有 IPC 唤醒其他核的功能）生成.out，然后经过工具链生成.dat,在通过烧写工具烧入 flash，拨码开关调整并断电重启，完成 boot。
5. 本文档主要描述拨码开关与寄存器的关系，boot paramter table 具体涵义，ddr configration table 的具体涵义，如果要快速实现 boot，参照 SPI_boot 操作指导，如果要了解更多 boot 相关以及多核 boot，可参考多核启动**研究**或者 TI boot loader 文档 sprugy5B 以及 bootloader 源码。
6. 另外要格外注意的是 evm 板卡上的 EEPROM 的 IBL 镜像文件存在没有被刷掉，否则无法正常引导 flash。
7. 知道工程里 L2 地址要写全局地址（0x1x8xxxxx）的形式，另外工程中核 0 有 IPC 触发中断
8. 如果是超大文件 boot，注意看 rmd 文件里的 length 是否够大，否则 hex6x 将无法转换全部的文件。

一． 配置 Device configuration。

667xEVM 板拨码开关

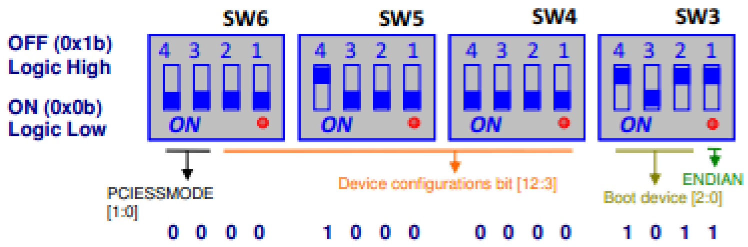


Figure 3.6: SW3, SW4, SW5, and SW6 default settings

拨码开关说明：

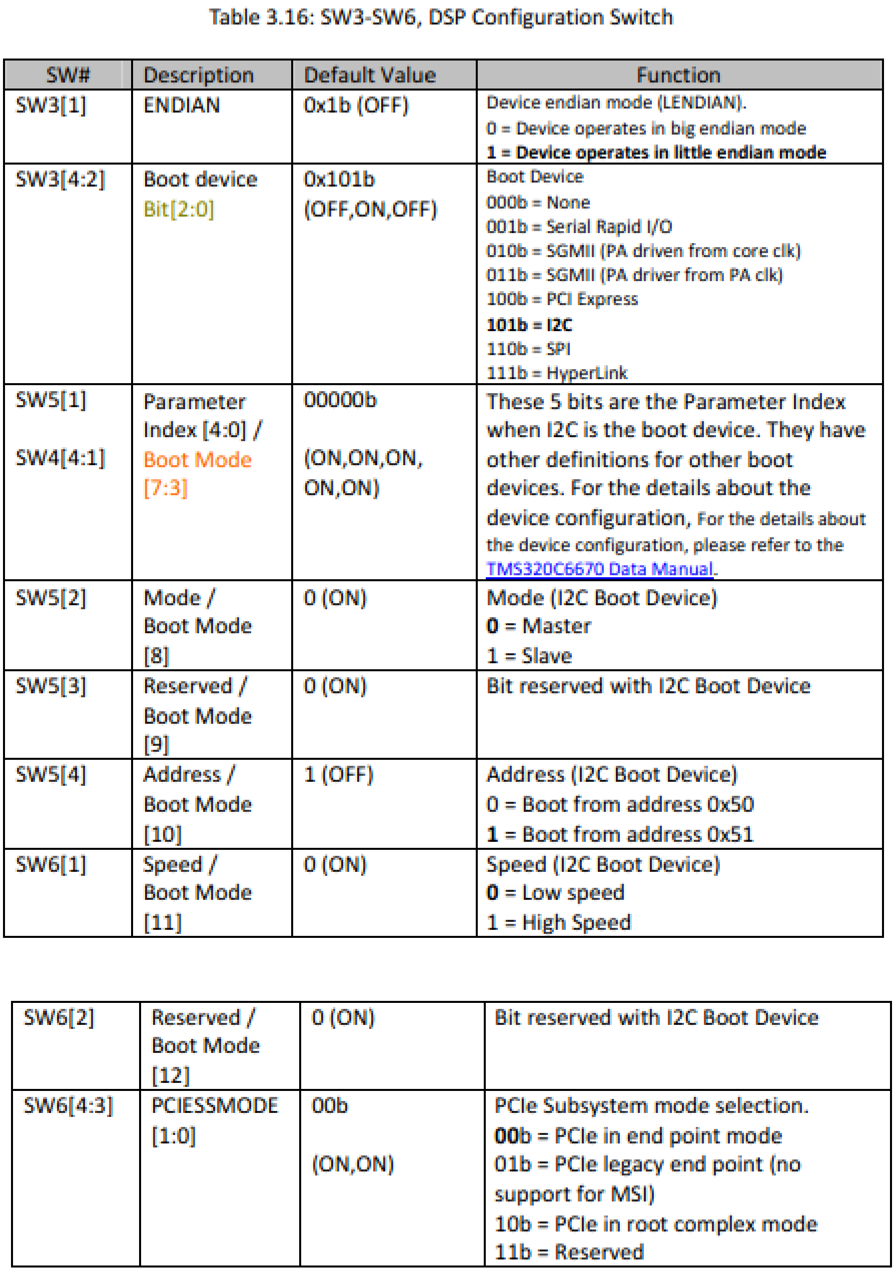


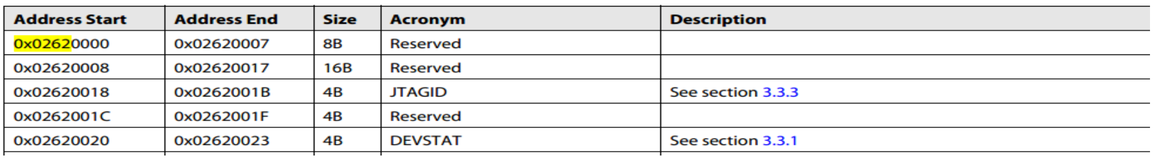
Table 3.16: SW3-SW6, DSP Configuration Switch

SW#	Description	Default Value	Function
SW3[1]	ENDIAN	0x1b (OFF)	Device endian mode (LENDIAN). 0 = Device operates in big endian mode **1 = Device operates in little endian mode**
SW3[4:2]	Boot device Bit[2:0]	0x101b (OFF,ON,OFF)	Boot Device 000b = None 001b = Serial Rapid I/O 010b = SGMII (PA driven from core clk) 011b = SGMII (PA driver from PA clk) 100b = PCI Express **101b = I2C** 110b = SPI 111b = HyperLink
SW5[1] SW4[4:1]	Parameter Index [4:0] / Boot Mode [7:3]	00000b (ON,ON,ON, ON,ON)	These 5 bits are the Parameter Index when I2C is the boot device. They have other definitions for other boot devices. For the details about the device configuration, For the details about the device configuration, please refer to the TMS320C6670 Data Manual.
SW5[2]	Mode / Boot Mode [8]	0 (ON)	Mode (I2C Boot Device) **0** = Master 1 = Slave
SW5[3]	Reserved / Boot Mode [9]	0 (ON)	Bit reserved with I2C Boot Device
SW5[4]	Address / Boot Mode [10]	1 (OFF)	Address (I2C Boot Device) 0 = Boot from address 0x50 **1** = Boot from address 0x51
SW6[1]	Speed / Boot Mode [11]	0 (ON)	Speed (I2C Boot Device) **0** = Low speed 1 = High Speed
SW6[2]	Reserved / Boot Mode [12]	0 (ON)	Bit reserved with I2C Boot Device
SW6[4:3]	PCIESSMODE [1:0]	00b (ON,ON)	PCIe Subsystem mode selection. **00**b = PCIe in end point mode 01b = PCIe legacy end point (no support for MSI) 10b = PCIe in root complex mode 11b = Reserved

SW4~SW6 因 boot 模式不同定义不一样

DEVSTAT 寄存器中存储了 boot 启动的相关信息

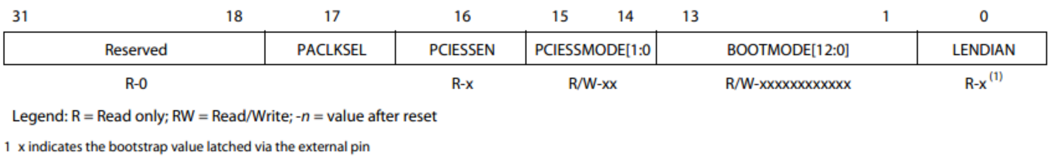
可见 DEVSTAT 寄存器的地址是 0x02620020,长度是 4 个字节



Address Start	Address End	Size	Acronym	Description
0x02620000	0x02620007	8B	Reserved	
0x02620008	0x02620017	16B	Reserved	
0x02620018	0x0262001B	4B	JTAGID	See section 3.3.3
0x0262001C	0x0262001F	4B	Reserved	
0x02620020	0x02620023	4B	DEVSTAT	See section 3.3.1

（拨码开关与 **DEVSTAT** 映射）**SW6~SW3 16 个 pin 对应的位就是 DEVSTAT 寄存器 bit16~0**

Figure 3-1 Device Status Register



31 18	17	16	15 14	13 1	0
Reserved	PACLKSEL	PCIESSEN	PCIESSMODE[1:0	BOOTMODE[12:0]	LENDIAN
R-0		R-x	R/W-xx	R/W-xxxxxxxxxxxx	R-x [1]

Legend: R = Read only; RW = Read/Write; -*n* = value after reset

1 x indicates the bootstrap value latched via the external pin

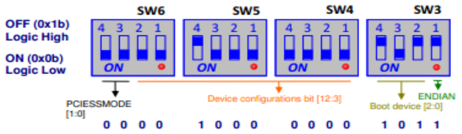
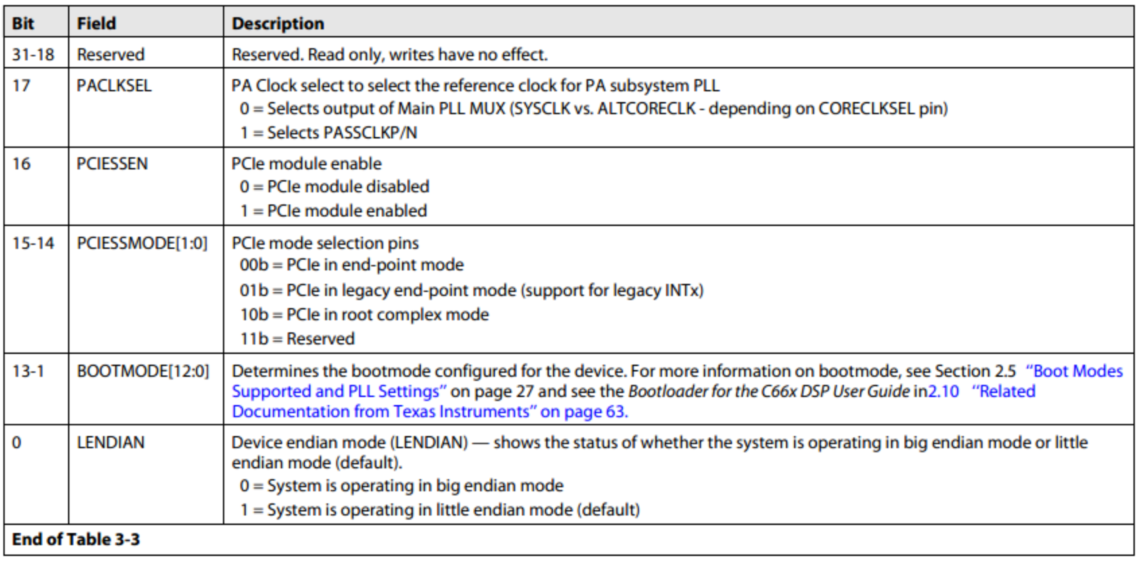


Figure 3.6: SW3, SW4, SW5, and SW6 default settings

以下是具体含义：

Table 3-3 Device Status Register Field Descriptions



Bit	Field	Description
31-18	Reserved	Reserved. Read only, writes have no effect.
17	PACLKSEL	PA Clock select to select the reference clock for PA subsystem PLL 0 = Selects output of Main PLL MUX (SYSCLK vs. ALTCORECLK - depending on CORECLKSEL pin) 1 = Selects PASSCLKP/N
16	PCIESSEN	PCIe module enable 0 = PCIe module disabled 1 = PCIe module enabled
15-14	PCIESSMODE[1:0]	PCIe mode selection pins 00b = PCIe in end-point mode 01b = PCIe in legacy end-point mode (support for legacy INTx) 10b = PCIe in root complex mode 11b = Reserved
13-1	BOOTMODE[12:0]	Determines the bootmode configured for the device. For more information on bootmode, see Section 2.5 "Boot Modes Supported and PLL Settings" on page 27 and see the *Bootloader for the C66x DSP User Guide* in2.10 "Related Documentation from Texas Instruments" on page 63.
0	LENDIAN	Device endian mode (LENDIAN) — shows the status of whether the system is operating in big endian mode or little endian mode (default). 0 = System is operating in big endian mode 1 = System is operating in little endian mode (default)
End of Table 3-3		

no boot

SW6	**SW5**	**SW4**	**SW3**
0011	0000	0000	0001

以 SPI boot 为例展开说明：

SW6	**SW5**	**SW4**	**SW3**		
0001	0100	0000	1101	off 1	on 0

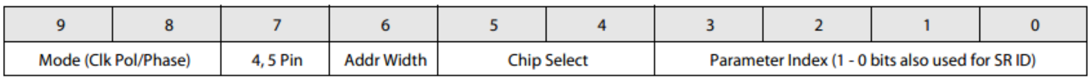
结合上表说明：

bit0： 小端模式

bit3~bit1 boot 模式（比如 110 是 SPI boot）

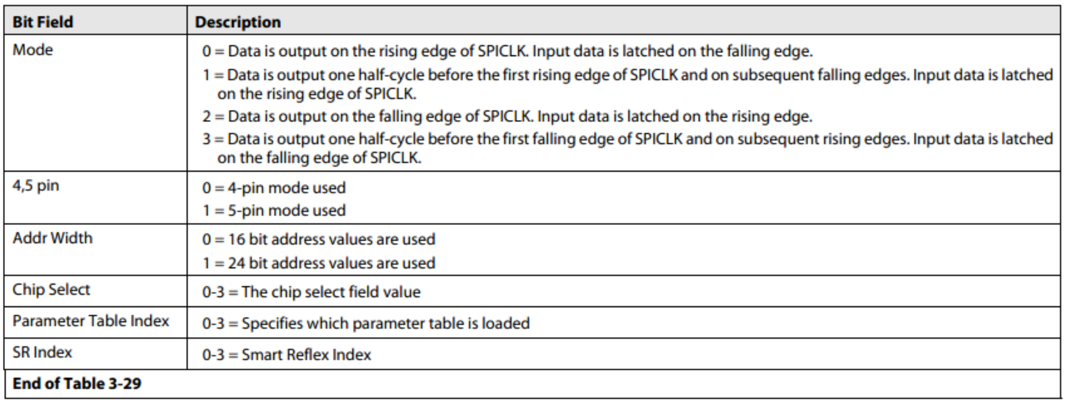
bit13~bit4 根据不同的 boot 模式而异，下表是 SPI boot 的这 10 个比特定义

Table 3-28 SPI Boot Mode Device Configuration



9	8	7	6	5	4	3	2	1	0
Mode (Clk Pol/Phase)		4, 5 Pin	Addr Width	Chip Select		Parameter Index (1 - 0 bits also used for SR ID)			

Table 3-29 SPI Boot Mode Device Configuration Description



Bit Field	Description
Mode	0 = Data is output on the rising edge of SPICLK. Input data is latched on the falling edge. 1 = Data is output one half-cycle before the first rising edge of SPICLK and on subsequent falling edges. Input data is latched on the rising edge of SPICLK. 2 = Data is output on the falling edge of SPICLK. Input data is latched on the rising edge. 3 = Data is output one half-cycle before the first falling edge of SPICLK and on subsequent rising edges. Input data is latched on the falling edge of SPICLK.
4,5 pin	0 = 4-pin mode used 1 = 5-pin mode used
Addr Width	0 = 16 bit address values are used 1 = 24 bit address values are used
Chip Select	0-3 = The chip select field value
Parameter Table Index	0-3 = Specifies which parameter table is loaded
SR Index	0-3 = Smart Reflex Index

End of Table 3-29

SR Index 是 00,

Parameter Table Index 是 00,

Chip Select 片选也是 00,

Addr Width 是 1，代表 24 bit address,

4,5pin mode 是 0，使用的是 4pin 模式,

Mode Clk phase 01， Data is output one half-cycle before…..

bit15~bit14: PCIe in end-point mode

至此 Device configuration 结束

二. 配置 boot parameter table

Boot 过程中，boot ROM 用到的地址空间为

start address： 0x008F 2DC0 //6670 或者 0x00872DC0//6678

end address ： 0x008F FFFF//6670 或者 0x0087FFFF//6678

length ： 0xD240 //52K

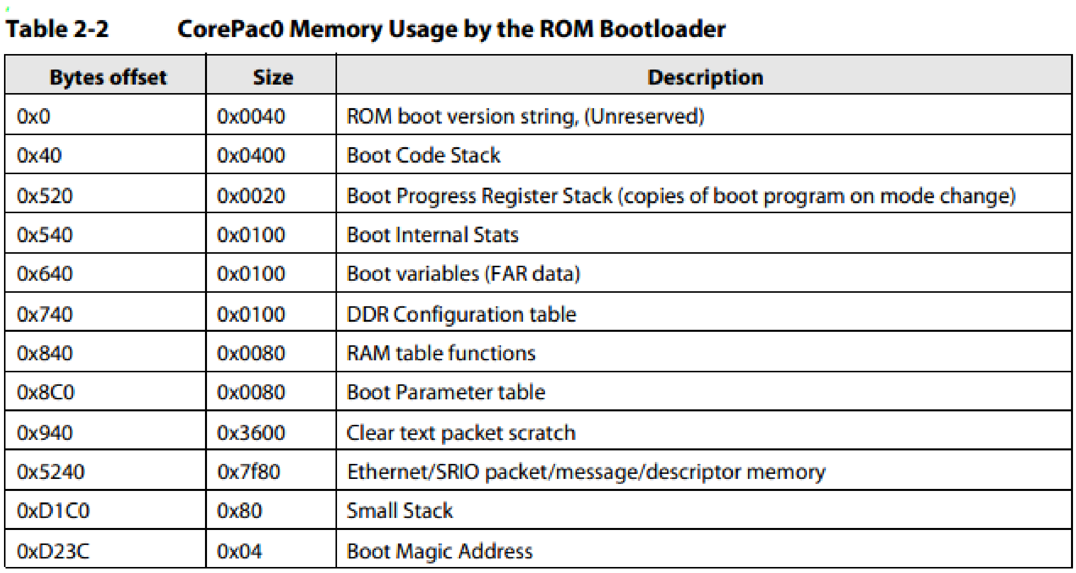


Table 2-2 CorePac0 Memory Usage by the ROM Bootloader

Bytes offset	Size	Description
0x0	0x0040	ROM boot version string, (Unreserved)
0x40	0x0400	Boot Code Stack
0x520	0x0020	Boot Progress Register Stack (copies of boot program on mode change)
0x540	0x0100	Boot Internal Stats
0x640	0x0100	Boot variables (FAR data)
0x740	0x0100	DDR Configuration table
0x840	0x0080	RAM table functions
0x8C0	0x0080	Boot Parameter table
0x940	0x3600	Clear text packet scratch
0x5240	0x7f80	Ethernet/SRIO packet/message/descriptor memory
0xD1C0	0x80	Small Stack
0xD23C	0x04	Boot Magic Address

说明：

1. no-boot 和 spi boot 以及 I2C boot 是不负责配置主 PLL 的，需要应用程序自己去写。

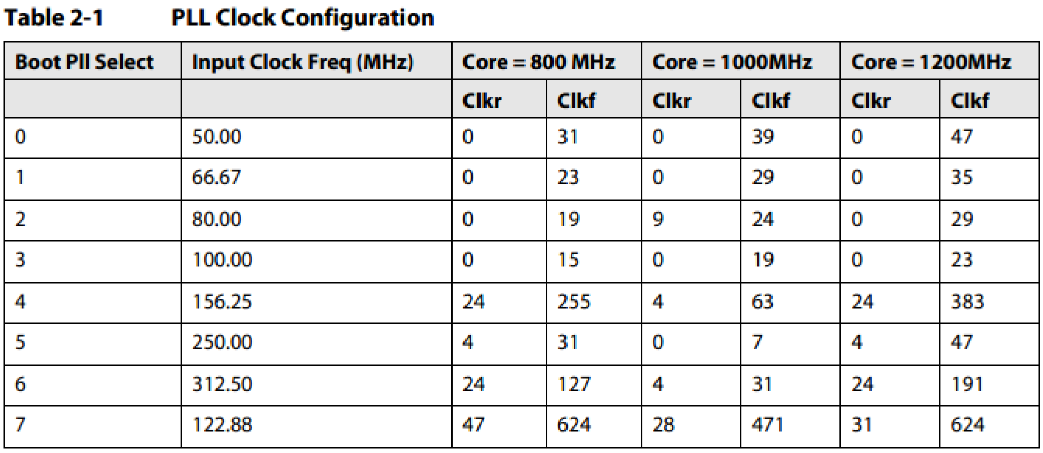


Table 2-1 PLL Clock Configuration

Boot Pll Select	Input Clock Freq (MHz)	Core = 800 MHz		Core = 1000MHz		Core = 1200MHz	
		Clkr	Clkf	Clkr	Clkf	Clkr	Clkf
0	50.00	0	31	0	39	0	47
1	66.67	0	23	0	29	0	35
2	80.00	0	19	9	24	0	29
3	100.00	0	15	0	19	0	23
4	156.25	24	255	4	63	24	383
5	250.00	4	31	0	7	4	47
6	312.50	24	127	4	31	24	191
7	122.88	47	624	28	471	31	624

2. L1D，L1P 被 boot ROM 自动配成全 cache，L2 自动配成全可寻址的 RAM
3. DDR 先被 DDR table 初始化，然后才会有数据 load 进 DDR，将会在第三章描述。

实际在被烧写文件中 parameter table 却只有 32 个字节 前 12 个是所有 boot 模式共用的，后 24 个字节是 SPI boot 专属

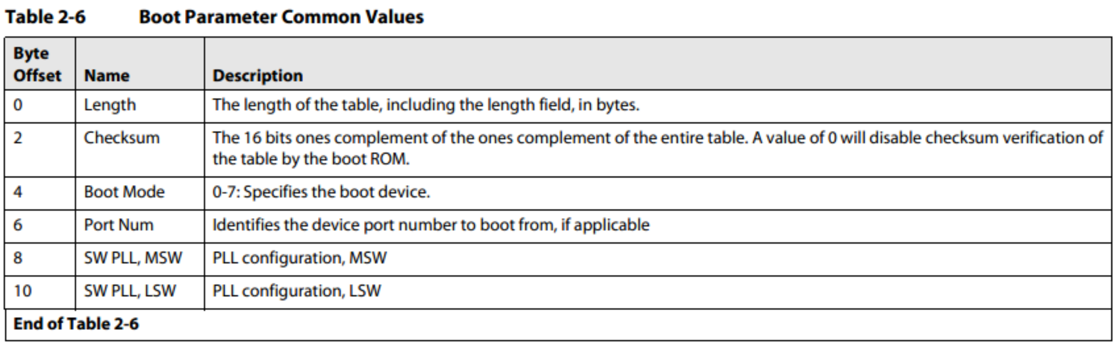


Table 2-6 Boot Parameter Common Values

Byte Offset	Name	Description
0	Length	The length of the table, including the length field, in bytes.
2	Checksum	The 16 bits ones complement of the ones complement of the entire table. A value of 0 will disable checksum verification of the table by the boot ROM.
4	Boot Mode	0-7: Specifies the boot device.
6	Port Num	Identifies the device port number to boot from, if applicable
8	SW PLL, MSW	PLL configuration, MSW
10	SW PLL, LSW	PLL configuration, LSW

End of Table 2-6

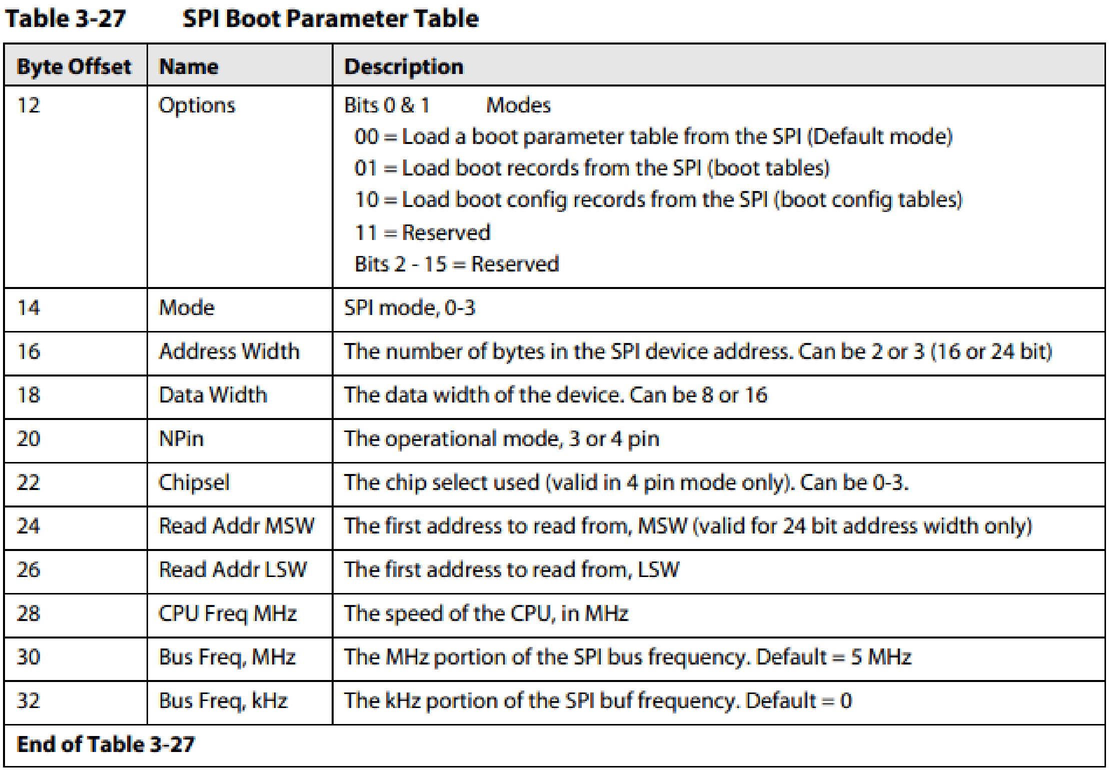
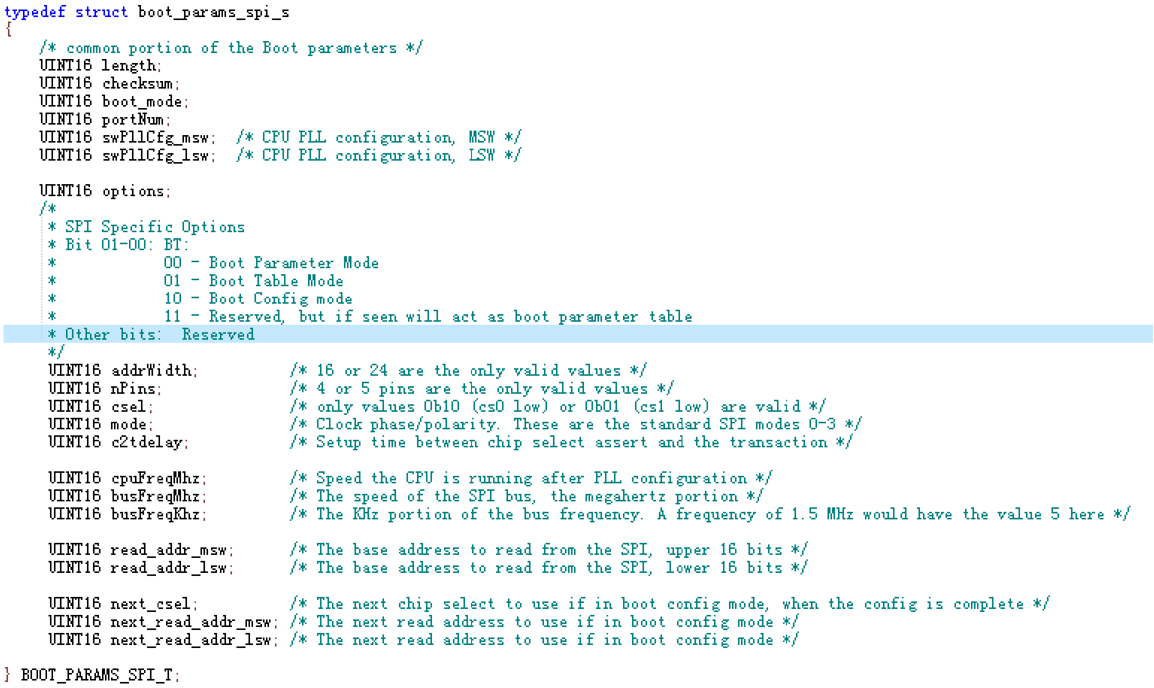


Table 3-27 SPI Boot Parameter Table

Byte Offset	Name	Description
12	Options	Bits 0 & 1 Modes 00 = Load a boot parameter table from the SPI (Default mode) 01 = Load boot records from the SPI (boot tables) 10 = Load boot config records from the SPI (boot config tables) 11 = Reserved Bits 2 - 15 = Reserved
14	Mode	SPI mode, 0-3
16	Address Width	The number of bytes in the SPI device address. Can be 2 or 3 (16 or 24 bit)
18	Data Width	The data width of the device. Can be 8 or 16
20	NPin	The operational mode, 3 or 4 pin
22	Chipsel	The chip select used (valid in 4 pin mode only). Can be 0-3.
24	Read Addr MSW	The first address to read from, MSW (valid for 24 bit address width only)
26	Read Addr LSW	The first address to read from, LSW
28	CPU Freq MHz	The speed of the CPU, in MHz
30	Bus Freq, MHz	The MHz portion of the SPI bus frequency. Default = 5 MHz
32	Bus Freq, kHz	The kHz portion of the SPI buf frequency. Default = 0

End of Table 3-27

另外要说的是，似乎 ROM bootloadre 的源代码和此表不符，下面是源代码的结构体：以 boot rom 源代码的结构体为准。



```
typedef struct boot_params_spi_s
{
    /* common portion of the Boot parameters */
    UINT16 length;
    UINT16 checksum;
    UINT16 boot_mode;
    UINT16 portNum;
    UINT16 swPllCfg_msw;  /* CPU PLL configuration, MSW */
    UINT16 swPllCfg_lsw;  /* CPU PLL configuration, LSW */

    UINT16 options;
    /*
     * SPI Specific Options
     * Bit 01-00: BT:
     *            00 - Boot Parameter Mode
     *            01 - Boot Table Mode
     *            10 - Boot Config mode
     *            11 - Reserved, but if seen will act as boot parameter table
     * Other bits:  Reserved
     */
     UINT16 addrWidth;          /* 16 or 24 are the only valid values */
     UINT16 nPins;              /* 4 or 5 pins are the only valid values */
     UINT16 csel;               /* only values 0b10 (cs0 low) or 0b01 (cs1 low) are valid */
     UINT16 mode;               /* Clock phase/polarity. These are the standard SPI modes 0-3 */
     UINT16 c2tdelay;           /* Setup time between chip select assert and the transaction */

     UINT16 cpuFreqMhz;         /* Speed the CPU is running after PLL configuration */
     UINT16 busFreqMhz;         /* The speed of the SPI bus, the megahertz portion */
     UINT16 busFreqKhz;         /* The KHz portion of the bus frequency. A frequency of 1.5 MHz would have the value 5 here */

     UINT16 read_addr_msw;      /* The base address to read from the SPI, upper 16 bits */
     UINT16 read_addr_lsw;      /* The base address to read from the SPI, lower 16 bits */

     UINT16 next_csel;          /* The next chip select to use if in boot config mode, when the config is complete */
     UINT16 next_read_addr_msw; /* The next read address to use if in boot config mode */
     UINT16 next_read_addr_lsw; /* The next read address to use if in boot config mode */

} BOOT_PARAMS_SPI_T;
```

对应上述结构体为下面文件：

此为 i2crom.ccs 文件

0x 0050 0000

Length checksum

0x 0032 0000 // bootmode 50 SPI boot portnum : 0

Bootmode portnum

这里 6670 和 6678 有所区别

（6670）

0x 4051 0502 //见下表 PLL configuration

swPllCfg_msw swPllCfg_lsw

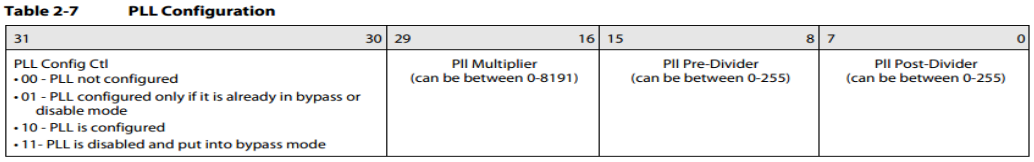


Table 2-7 PLL Configuration

31 30	29 16	15 8	7 0
PLL Config Ctl • 00 - PLL not configured • 01 - PLL configured only if it is already in bypass or disable mode • 10 - PLL is configured • 11- PLL is disabled and put into bypass mode	Pll Multiplier (can be between 0-8191)	Pll Pre-Divider (can be between 0-255)	Pll Post-Divider (can be between 0-255)

31:30 01 主 pll 处于 bypassmode

29:16 00000001010001 Pll Multipiler 81

15:8 00000101 pre-divider 5

7:0 00000010 post divider 2

所以 cpu 主频应该是 122.88 * 81 / 5 / 2 = 995.328

（6678）

0x 4013 0002 //见下表 PLL configuration

swPllCfg_msw swPllCfg_lsw

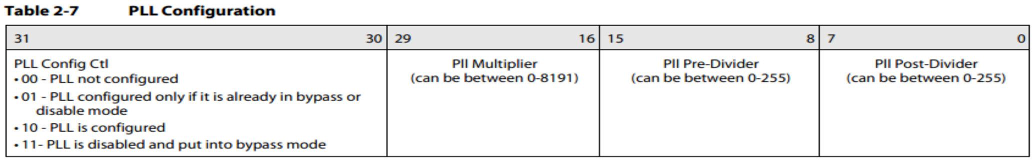


Table 2-7 PLL Configuration

31 30	29 16	15 8	7 0
PLL Config Ctl • 00 - PLL not configured • 01 - PLL configured only if it is already in bypass or disable mode • 10 - PLL is configured • 11- PLL is disabled and put into bypass mode	Pll Multiplier (can be between 0-8191)	Pll Pre-Divider (can be between 0-255)	Pll Post-Divider (can be between 0-255)

31:30 01 主 pll 处于 bypassmode

29:16 00 0000 0001 0011 Pll Multipiler 19

15:8 0000 0000 pre-divider 0

7:0 0000 0010 post divider 2

所以 cpu 主频应该是 100 * 19 / 2 = 950

0x 0001 0018 //boot table mode addrWidth = 24

options addrWidth

0x 0004 0000 // 4 pin flash，片选选择了 0 似乎不合代码里的 valid 值

npins csel

```
0x        0001                0000     //  待查证
           mode              c2tdelay
 --------------------------------------------------------------------------------------------------------------------
0x        03e8                0000    // 1000 MHZ cpu
      cpuFreqMhz          busFreqMhz
--------------------------------------------------------------------------------------------------------------------
0x        01f4                 0000          //500K  bus  rate  ；  后四位原来是  0x0051,因为  boot
                                                 paramter table 不全所以改为了 0000
        bhsFreqKhz          readaddrMsw
--------------------------------------------------------------------------------------------------------------------
0x        0400                0000      //read address is 0x00000400  即 1Kbytes  的地方， .dat 文
                                          件里 boot table  确实从第 256 行开始
      readaddrLsw          next_csel
--------------------------------------------------------------------------------------------------------------------
0x        0000                0000
next_Read_addr_msw    next_Read_addr_lsw
--------------------------------------------------------------------------------------------------------------------
```

三．配置 DDR　configuration table

6678:

DDR3 boot table：

```
00 00 00 70 00 87 35 00 02 42 80 F5 00 00 00 00 00 00 00 1C 00 00 00 02
63 06 2A 32 00 00 00 00 00 00 14 50 11 13 78 3C 30 71 7F E3 55 9F 86 AF
00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00
00 00 00 00 70 07 32 14 00 00 00 00 00 10 01 0F 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 03 05
```

将上段代码加在.btbl 文件的前四个字节之后（main 函数入口点）
再次粘入 DDR table 配置表

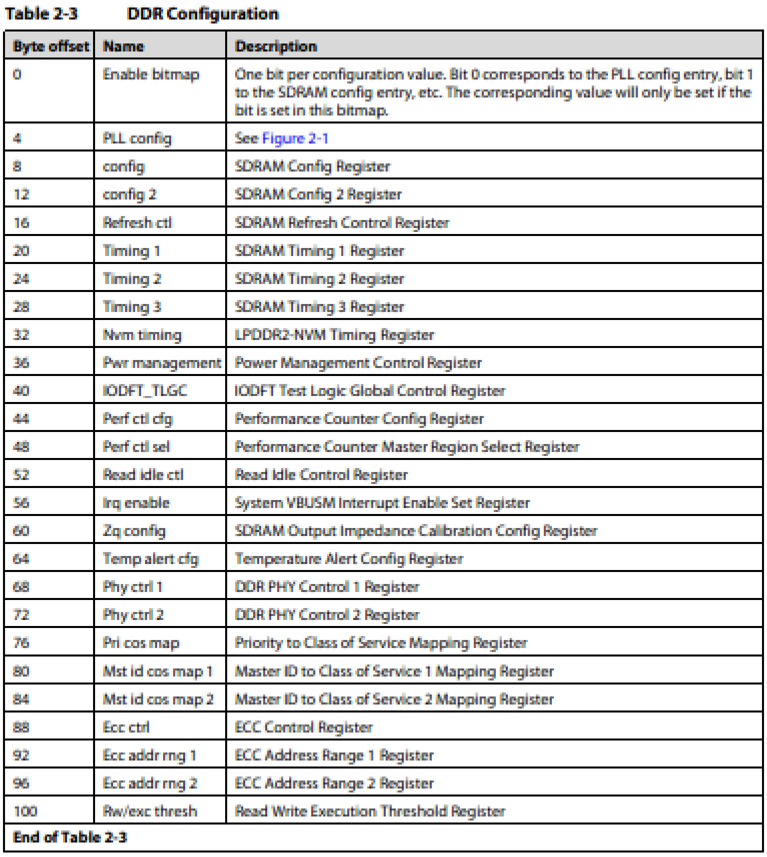
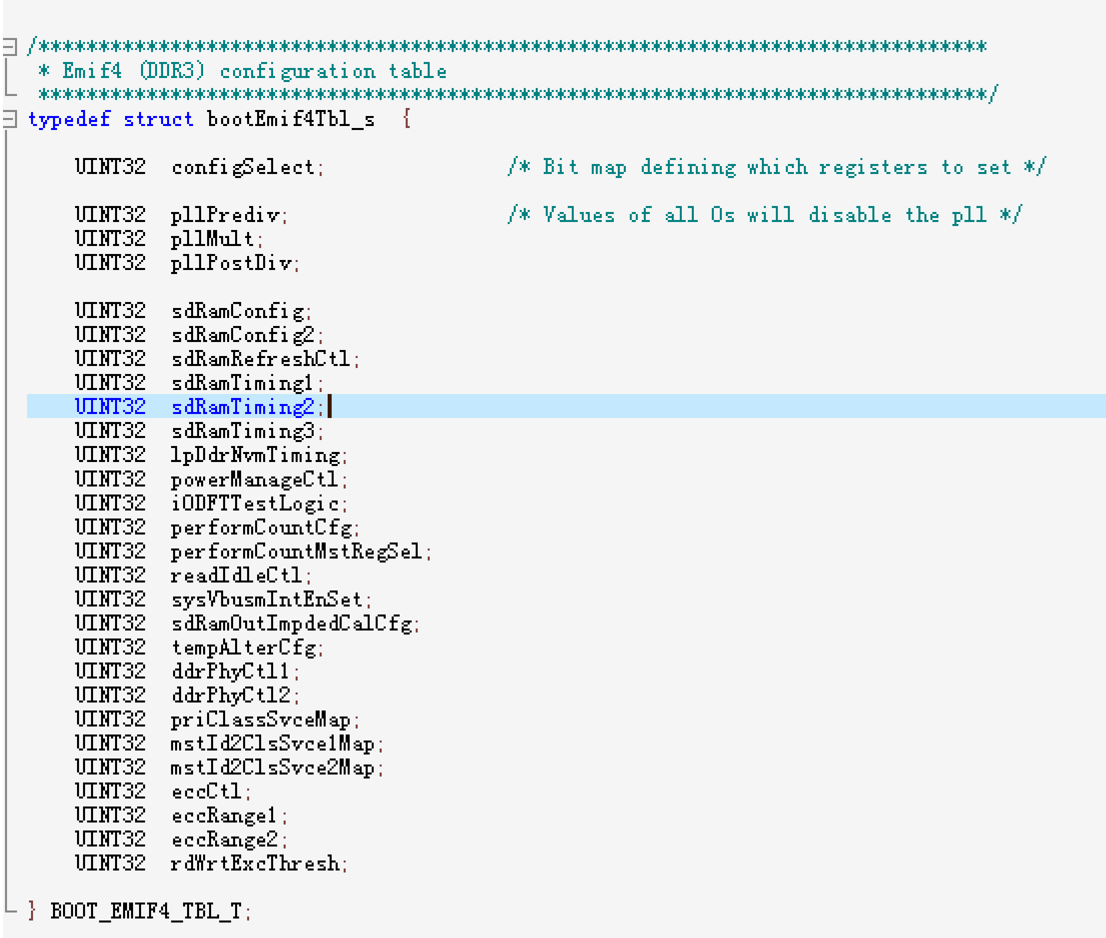


Table 2-3　　DDR Configuration

Byte offset	Name	Description
0	Enable bitmap	One bit per configuration value. Bit 0 corresponds to the PLL config entry, bit 1 to the SDRAM config entry, etc. The corresponding value will only be set if the bit is set in this bitmap.
4	PLL config	See Figure 2-1
8	config	SDRAM Config Register
12	config 2	SDRAM Config 2 Register
16	Refresh ctl	SDRAM Refresh Control Register
20	Timing 1	SDRAM Timing 1 Register
24	Timing 2	SDRAM Timing 2 Register
28	Timing 3	SDRAM Timing 3 Register
32	Nvm timing	LPDDR2-NVM Timing Register
36	Pwr management	Power Management Control Register
40	IODFT_TLGC	IODFT Test Logic Global Control Register
44	Perf ctl cfg	Performance Counter Config Register
48	Perf ctl sel	Performance Counter Master Region Select Register
52	Read idle ctl	Read Idle Control Register
56	Irq enable	System VBUSM Interrupt Enable Set Register
60	Zq config	SDRAM Output Impedance Calibration Config Register
64	Temp alert cfg	Temperature Alert Config Register
68	Phy ctrl 1	DDR PHY Control 1 Register
72	Phy ctrl 2	DDR PHY Control 2 Register
76	Pri cos map	Priority to Class of Service Mapping Register
80	Mst id cos map 1	Master ID to Class of Service 1 Mapping Register
84	Mst id cos map 2	Master ID to Class of Service 2 Mapping Register
88	Ecc ctrl	ECC Control Register
92	Ecc addr rng 1	ECC Address Range 1 Register
96	Ecc addr rng 2	ECC Address Range 2 Register
100	Rw/exc thresh	Read Write Execution Threshold Register

End of Table 2-3

注意：以上配置表完全和 src code 对不上，不要参考对照配置，应按照以下 src code 里的源代码为准,结合论坛上关于 PllPrediv 三个 word 的说法，有以下表



```
/*************************************************************************
 * Emif4 (DDR3) configuration table
 *************************************************************************/
typedef struct bootEmif4Tbl_s  {

    UINT32   configSelect;              /* Bit map defining which registers to set */

    UINT32   pllPrediv;                 /* Values of all 0s will disable the pll */
    UINT32   pllMult;
    UINT32   pllPostDiv;

    UINT32   sdRamConfig;
    UINT32   sdRamConfig2;
    UINT32   sdRamRefreshCtl;
    UINT32   sdRamTiming1;
    UINT32   sdRamTiming2;
    UINT32   sdRamTiming3;
    UINT32   lpDdrNvmTiming;
    UINT32   powerManageCtl;
    UINT32   iODFTTestLogic;
    UINT32   performCountCfg;
    UINT32   performCountMstRegSel;
    UINT32   readIdleCtl;
    UINT32   sysVbusmIntEnSet;
    UINT32   sdRamOutImpdedCalCfg;
    UINT32   tempAlterCfg;
    UINT32   ddrPhyCtl1;
    UINT32   ddrPhyCtl2;
    UINT32   priClassSvceMap;
    UINT32   mstId2ClsSvce1Map;
    UINT32   mstId2ClsSvce2Map;
    UINT32   eccCtl;
    UINT32   eccRange1;
    UINT32   eccRange2;
    UINT32   rdWrtExcThresh;

} BOOT_EMIF4_TBL_T;
```

```
00 00 00 70    //???
00 87 35 00    //where to load ddr table to L2

02 42 80 F5    //config select

00 00 00 00    //pll Prediv
00 00 00 1C    //pll Mul
00 00 00 02    //pll post div

63 06 2A 32    //sd ram config
00 00 00 00    //sdram config 2
00 00 14 50    //sdram fresh cntrl
11 13 78 3C    //sdram timing 1
30 71 7F E3    //sdram timing 2
```

```
55 9F 86 AF    //sdram timing 3
00 00 00 00
00 00 00 00
00 00 00 00
00 01 00 00 //perform count config
00 00 00 00
00 00 00 00
00 00 00 00
70 07 32 14 //sdRamoutImpedCalcfg
00 00 00 00
00 10 01 0F //ddr phy control1
00 00 00 00
00 00 00 00
00 00 00 00
00 00 00 00
10 00 00 00 //ecc cntrl
00 00 00 00
00 00 00 00
00 00 03 05//rdWrtExcThresh
```

上述bootloader里关于ddr3的对应文档为DDR3 memory controller user guide，Table4-1

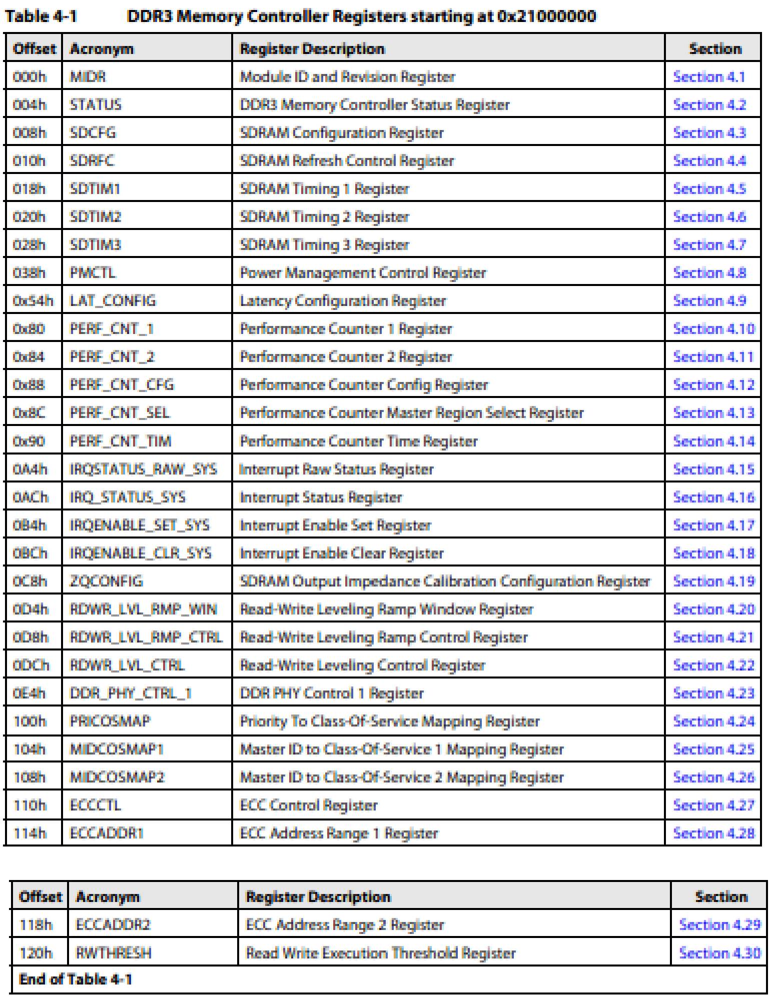


Table 4-1 DDR3 Memory Controller Registers starting at 0x21000000

Offset	Acronym	Register Description	Section
000h	MIDR	Module ID and Revision Register	Section 4.1
004h	STATUS	DDR3 Memory Controller Status Register	Section 4.2
008h	SDCFG	SDRAM Configuration Register	Section 4.3
010h	SDRFC	SDRAM Refresh Control Register	Section 4.4
018h	SDTIM1	SDRAM Timing 1 Register	Section 4.5
020h	SDTIM2	SDRAM Timing 2 Register	Section 4.6
028h	SDTIM3	SDRAM Timing 3 Register	Section 4.7
038h	PMCTL	Power Management Control Register	Section 4.8
0x54h	LAT_CONFIG	Latency Configuration Register	Section 4.9
0x80	PERF_CNT_1	Performance Counter 1 Register	Section 4.10
0x84	PERF_CNT_2	Performance Counter 2 Register	Section 4.11
0x88	PERF_CNT_CFG	Performance Counter Config Register	Section 4.12
0x8C	PERF_CNT_SEL	Performance Counter Master Region Select Register	Section 4.13
0x90	PERF_CNT_TIM	Performance Counter Time Register	Section 4.14
0A4h	IRQSTATUS_RAW_SYS	Interrupt Raw Status Register	Section 4.15
0ACh	IRQ_STATUS_SYS	Interrupt Status Register	Section 4.16
0B4h	IRQENABLE_SET_SYS	Interrupt Enable Set Register	Section 4.17
0BCh	IRQENABLE_CLR_SYS	Interrupt Enable Clear Register	Section 4.18
0C8h	ZQCONFIG	SDRAM Output Impedance Calibration Configuration Register	Section 4.19
0D4h	RDWR_LVL_RMP_WIN	Read-Write Leveling Ramp Window Register	Section 4.20
0D8h	RDWR_LVL_RMP_CTRL	Read-Write Leveling Ramp Control Register	Section 4.21
0DCh	RDWR_LVL_CTRL	Read-Write Leveling Control Register	Section 4.22
0E4h	DDR_PHY_CTRL_1	DDR PHY Control 1 Register	Section 4.23
100h	PRICOSMAP	Priority To Class-Of-Service Mapping Register	Section 4.24
104h	MIDCOSMAP1	Master ID to Class-Of-Service 1 Mapping Register	Section 4.25
108h	MIDCOSMAP2	Master ID to Class-Of-Service 2 Mapping Register	Section 4.26
110h	ECCCTL	ECC Control Register	Section 4.27
114h	ECCADDR1	ECC Address Range 1 Register	Section 4.28
118h	ECCADDR2	ECC Address Range 2 Register	Section 4.29
120h	RWTHRESH	Read Write Execution Threshold Register	Section 4.30
End of Table 4-1			

6670:

6670 一切尽与 6678 相同，就是在 L2 里放置 ddr configuration table 的地方不一样

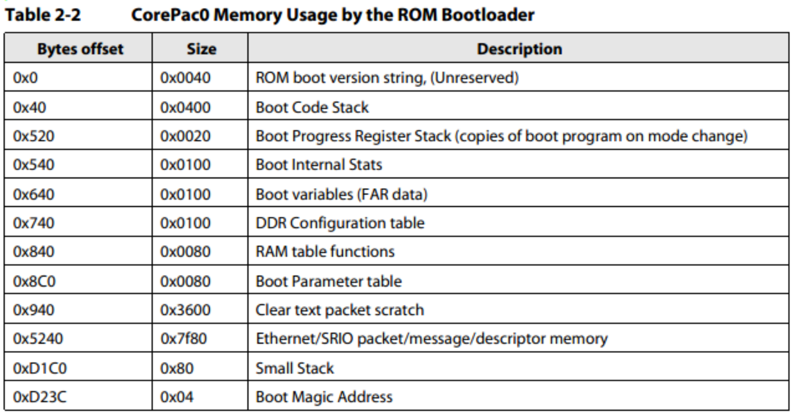


Table 2-2 CorePac0 Memory Usage by the ROM Bootloader

Bytes offset	Size	Description
0x0	0x0040	ROM boot version string, (Unreserved)
0x40	0x0400	Boot Code Stack
0x520	0x0020	Boot Progress Register Stack (copies of boot program on mode change)
0x540	0x0100	Boot Internal Stats
0x640	0x0100	Boot variables (FAR data)
0x740	0x0100	DDR Configuration table
0x840	0x0080	RAM table functions
0x8C0	0x0080	Boot Parameter table
0x940	0x3600	Clear text packet scratch
0x5240	0x7f80	Ethernet/SRIO packet/message/descriptor memory
0xD1C0	0x80	Small Stack
0xD23C	0x04	Boot Magic Address

再次粘入 table2-2，看相对偏移量 0x740 便可知，因为初始地址 6670 与 6678 不同，所以 ddr configuration table 的地址也不一样，于是有

DDR3 boot table：

```
00 00 00 70 00 8F 35 00 02 42 80 F5 00 00 00 00 00 00 00 1C 00 00 00 02
63 06 2A 32 00 00 00 00 00 00 14 50 11 13 78 3C 30 71 7F E3 55 9F 86 AF
00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00
00 00 00 00 70 07 32 14 00 00 00 00 00 10 01 0F 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 03 05
```