MSC

软件部 --- 邵望权

MSC

- MMC 相关
- 硬件电路
- 协议 ---SD 卡协议为主
- 硬件逻辑
- 软件实现 ---MMC 子系统

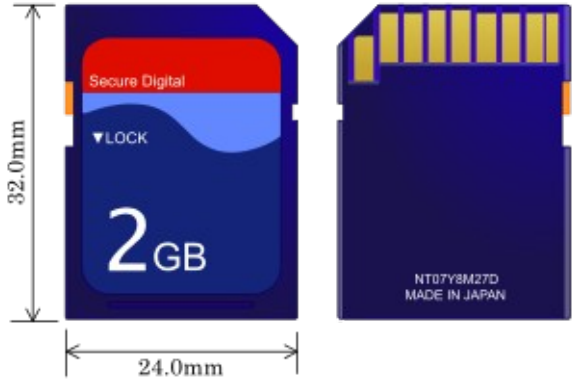
MMC 相关

- MMC （ MultiMediaCard ）
- MSC （ Mobile Storage Controller ）
- SD （ Secure Digital ）
  - SDSC （ <=2GB ）
  - SDHC （ >2Gb, <=32GB ）
  - SDXC （ >32GB, <= 2TB ）
- eMMC （ embedded MMC ）
- SDIO （ Secure Digital Input and Output ）

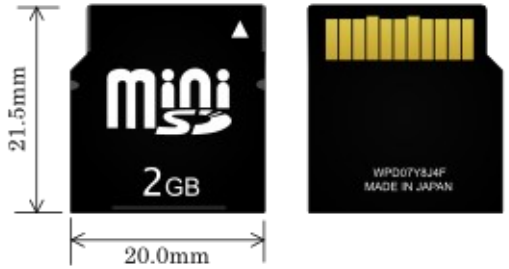
- 详情参考： https://en.wikipedia.org/wiki/Secure_Digital

规格

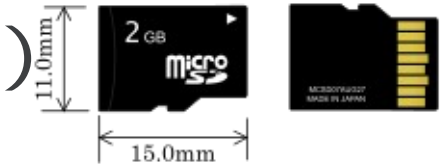
SD 卡



miniSD



microSD---（ TF Card ）

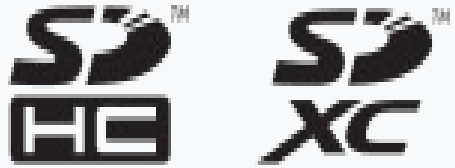


BUS 接口

- BUS
  - SPI 模式
  - SD 模式
  - UHS 模式
- UHS（ **Ultra High Speed** ）

类型	协议
UHS-I	SD version 3.01
UHS-II	SD version 4.0
UHS-III	SD version 6.0

速度模式

Bus interface	Card logo	Bus logo	Bus speed	Spec version
Default Speed		—	12.5 MByte/s	1.01
High Speed			25 MByte/s	2.00
UHS-I		I	12.5 MByte/s (SDR12) 25 MByte/s (SDR25) 50 MByte/s (SDR50,DDR50) 104 MByte/s (SDR104)	3.01
UHS-II		II	156 MByte/s (FD156) 312 MByte/s (HD312)	4.00/4.10 (X2000)
UHS-III			312 MByte/s (FD312) 624 MByte/s (FD624)	6.0

速度等级

Minimum sequential writing speed	Speed Class	UHS Speed Class	Video Speed Class	Application Performance Class	Application
2 MB/s	Class 2 (C2)	-	-	-	SD video recording
4 MB/s	Class 4 (C4)	-	-	-	High-definition video (HD) recording including Full HD (from 720p to 1080p/1080i)
6 MB/s	Class 6 (C6)	-	Class 6 (V6)	-	
10 MB/s	Class 10 (C10)	Class 1 (U1)	Class 10 (V10)	Class 1 (A1)	Full HD (1080p) video recording and consecutive recording of HD stills (High Speed bus, Class C10), real-time broadcasts and large HD video files (UHS bus, Classes U1 and V10) Running applications from the memory card (Class A1 - minimum 1500 read / 500 write operations per second)
30 MB/s	-	Class 3 (U3)	Class 30 (V30)	-	1080p and 4K video files at 60/120 fps (UHS bus)
60 MB/s	-	-	Class 60 (V60)	-	8K video files at 60/120 fps (UHS bus)
90 MB/s	-	-	Class 90 (V90)	-	

Card


SAMSUNG
8
micro SD HC


SanDisk
Ultra
64GB
micro SD XC I


Lexar
1000x
32GB
micro SD HC II

USH-III

BUS 与 Card

	SDSC card	SDHC card	SDHC UHS-I card	SDHC UHS-II card	SDXC card	SDXC UHS-I card	SDXC UHS-II card	SDIO card
SDSC slot	Yes	No	No	No	No	No	No	No
SDHC slot	Yes	Yes	Yes	Yes[]	No	No	No	No
SDHC UHS-I slot	Yes	Yes	Yes	Yes	No	No	No	No
SDHC UHS-II slot	Yes	Yes	Yes	Yes	No	No	No	No
SDXC slot	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
SDXC UHS-I slot	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
SDXC UHS-II slot	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No
SDIO slot	Varies	Varies	Varies	Varies	Varies	Varies	Varies	Yes

硬件电路

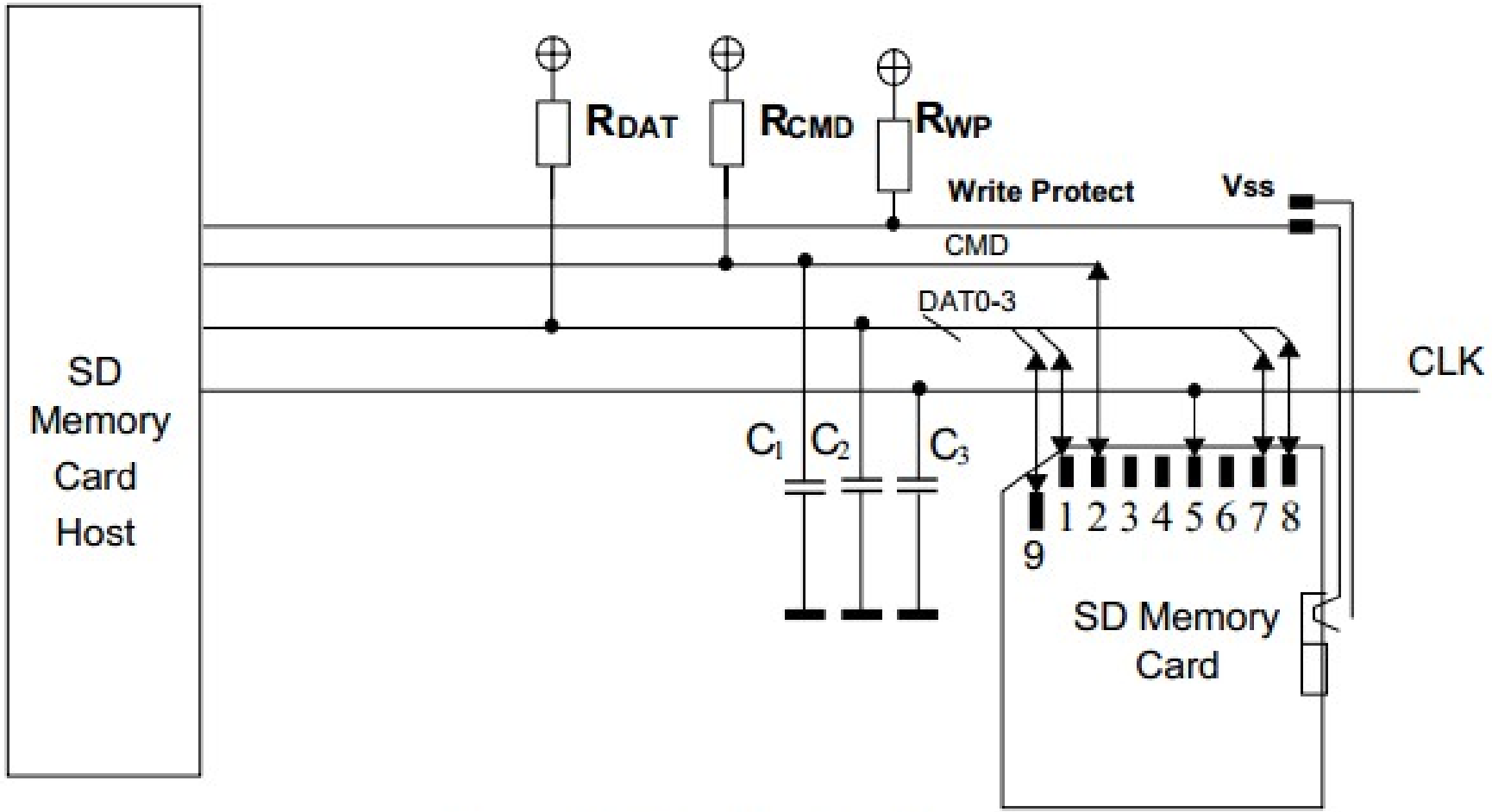


Figure 6-1: Bus Circuitry Diagram

硬件说明

SD pin	microSD pin	Name	I/O	Logic	Description
1	2	DAT3	I/O	PP	SD Serial Data 3
2	3	CMD	I/O	PP, OD	Command, Response
3		VSS	S	S	Ground
4	4	VDD	S	S	Power
5	5	CLK	I	PP	Serial clock
6	6	VSS	S	S	Ground
7	7	DAT0	I/O	PP	SD Serial Data 0
8	8	DAT1 nIRQ	I/O O	PP OD	SD Serial Data 1 (memory cards) **Interrupt Period** (SDIO cards share pin via protocol)
9	1	DAT2	I/O	PP	SD Serial Data 2

注：

数据方向： I = Input, O = Output.
PP = Push-Pull logic （上拉）
OD = Open-Drain logic （开漏）
S = Power Supply

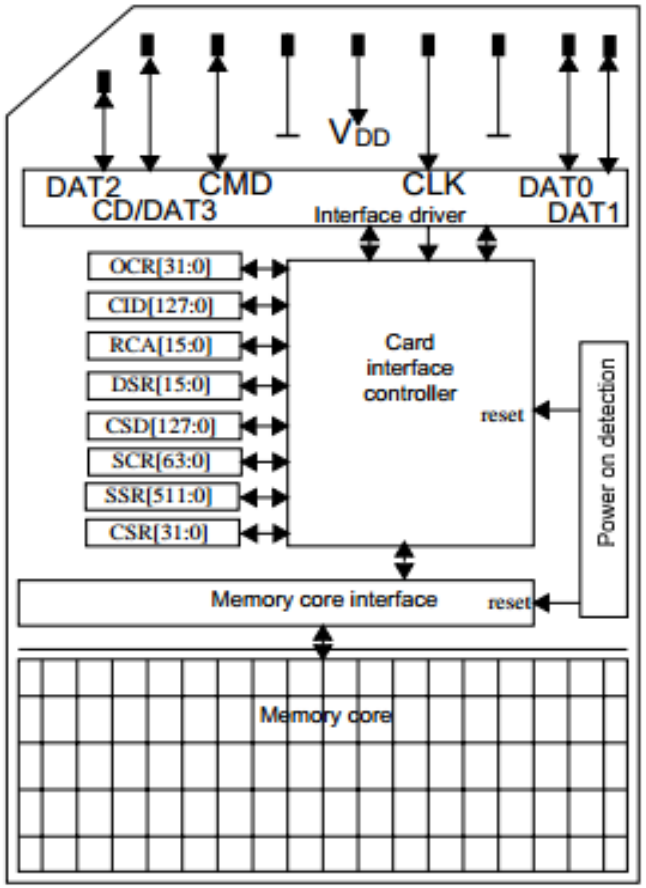


Figure 3-12: SD Memory Card Architecture

Name	Width	Description
CID	128	Card identification number; card individual number for identification (See 5.2). **Mandatory.**
RCA[1]	16	Relative card address; local system address of a card, dynamically suggested by the card and approved by the host during initialization (See 5.4). **Mandatory.**
DSR	16	Driver Stage Register; to configure the card's output drivers (See 5.5). **Optional.**
CSD	128	Card Specific Data; information about the card operation conditions (See 5.3). **Mandatory**
SCR	64	SD Configuration Register; information about the SD Memory Card's Special Features capabilities (See 5.6). **Mandatory**
OCR	32	Operation conditions register (See 5.1). **Mandatory.**
SSR	512	SD Status; information about the card proprietary features (See 4.10.2). **Mandatory**
CSR	32	Card Status; information about the card status (See 4.10.1). **Mandatory**

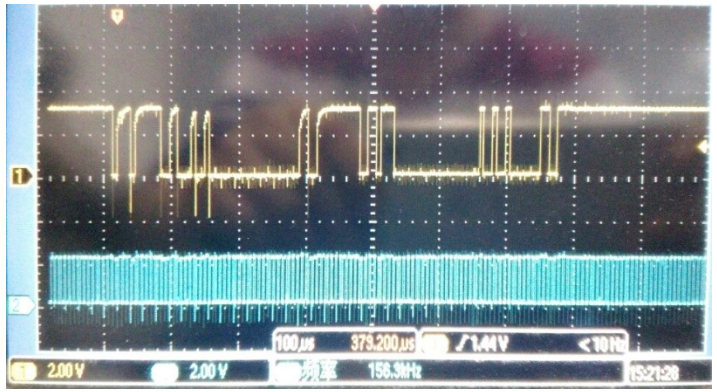
(1) RCA register is not used (available) in SPI mode

Table 3-2: SD Memory Card Registers

电压范围

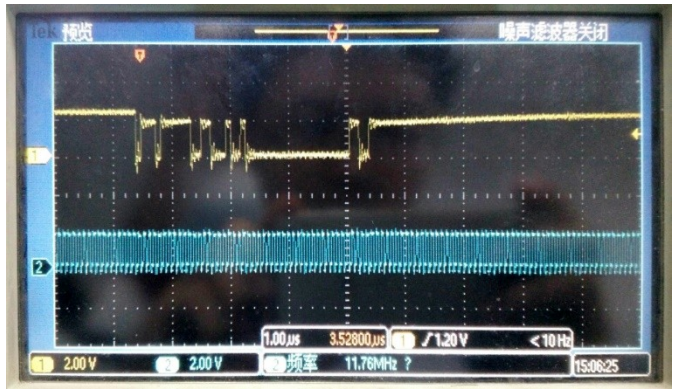
Parameter	Symbol	Min	Max	Unit	Remark
Supply Voltage	V_{DD}	2.7	3.6	V	
Output High Voltage	V_{OH}	0.75*V_{DD}		V	I_{OH}=-100uA $V_{DD\ min}$
Output Low Voltage	V_{OL}		0.125*V_{DD}	V	I_{OL} = 100uA $V_{DD\ min}$
Input High Voltage	V_{IH}	0.625*V_{DD}	V_{DD}+0.3	V	
Input Low Voltage	V_{IL}	V_{SS}-0.3	0.25 *V_{DD}	V	
Power Up Time			250	ms	From 0V to $V_{DD\ min}$

Table 6-2: Threshold Level for High Voltage



正常波形

上拉电阻大引起，但可以正常工作



上拉电阻过大，导致电压无法在规定时间（ 2 个时钟周期）内达到一定值

上拉电阻的范围： 10 ~ 100 kΩ

协议 ---SD 卡协议为主

- 协议：
  - CMD
  - CMD + DATA
- CMD 类型：
  - Broadcast commands (bc), no response
  - Broadcast commands with response (bcr)
  - Addressed (point-to-point) commands (ac)
  - Addressed (point-to-point) data transfer commands (adtc)

Command Class

	Card Command Class (CCC)	0	1	2	3	4	5	6	7	8	9	10	11
Supported commands	class description	basic	reserved	block read	reserved	block write	erase	write pro-tec-tion	lock card	appli-cation spe-cific	I/O mode	switch	reserved
CMD0	Mandatory	+											
CMD2	Mandatory	+											
CMD3	Mandatory	+											
CMD4	Mandatory	+											
CMD5	Optional										+		
CMD6[2]	Mandatory											+	
CMD7	Mandatory	+											
CMD8[3]	Mandatory	+											
CMD9	Mandatory	+											
CMD10	Mandatory	+											
CMD12	Mandatory	+											
CMD13	Mandatory	+											
CMD15	Mandatory	+											
CMD16	Mandatory			+		+			+				
CMD17	Mandatory			+									

Command

- 常规命令：

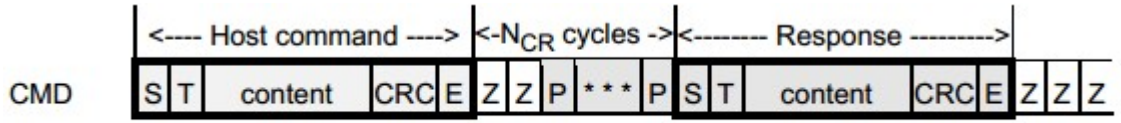


Figure 4-14: SEND_RELATIVE_ADDR Timing

- 读： CMD17/18

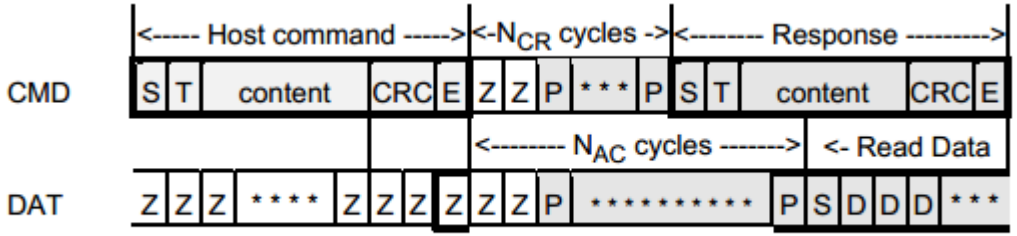


Figure 4-18: Timing of Single Block Read Command

- 写： CMD24/25

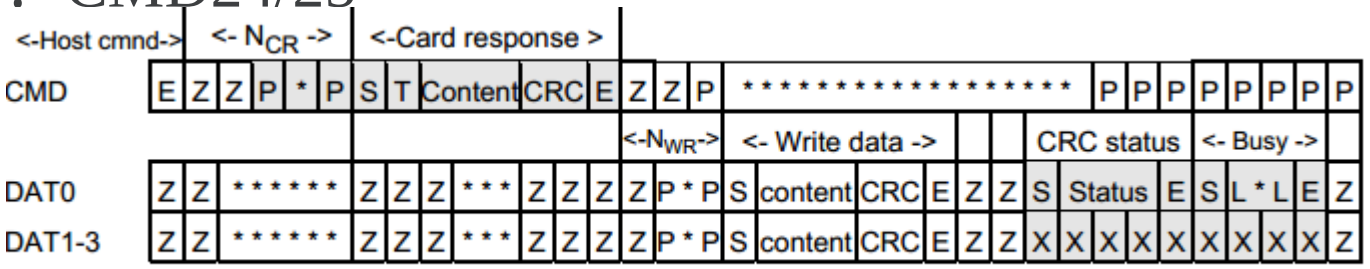


Figure 4-21: Timing of Single Block Write Command

Parameter	Min	Max	Unit
N_{CR}	2	64	clock cycles
N_{ID}	5	5	clock cycles
N_{AC} [1]	2	-	clock cycles
N_{RC}	8	-	clock cycles
N_{CC}	8	-	clock cycles
N_{WR}	2	-	clock cycles

1) The maximum read access time for a Standard Capacity SD Memory Card shall be calculated by host as follows:
Nac(max)= 100 ((TAAC * fpp) + (100 * NSAC)) ;
fpp is the interface clock rate and TAAC & NSAC are given in the CSD (Chapter 5.3).
Details of read, write, and erase timeouts are described in 4.6.2
In the case of a High Capacity SD Memory Card, a fixed value (100 ms) shall be used for the maximum read access time.

Table 4-47: Timing Values

写

- 在写的过程中由于控制器需要等到卡将数据全部写完，才视一次传输完成。
- 而在卡写的过程中，只有数据完全写入后，标志数据传输完成的 busy 位将在 DATA0 返回。同时返回的还有此次写数据后的状态 status(CRC 校验值）。如果 CRC 的校验值大于“ 010”, 将代表数据传输失败。
- CRC Status
  - “010” —— 数据被接受写入卡中
  - “101” —— 由于 CRC 错误，数据不被卡接受
  - “110” —— 由于写错误，数据不被卡接受

SD 初始化流程

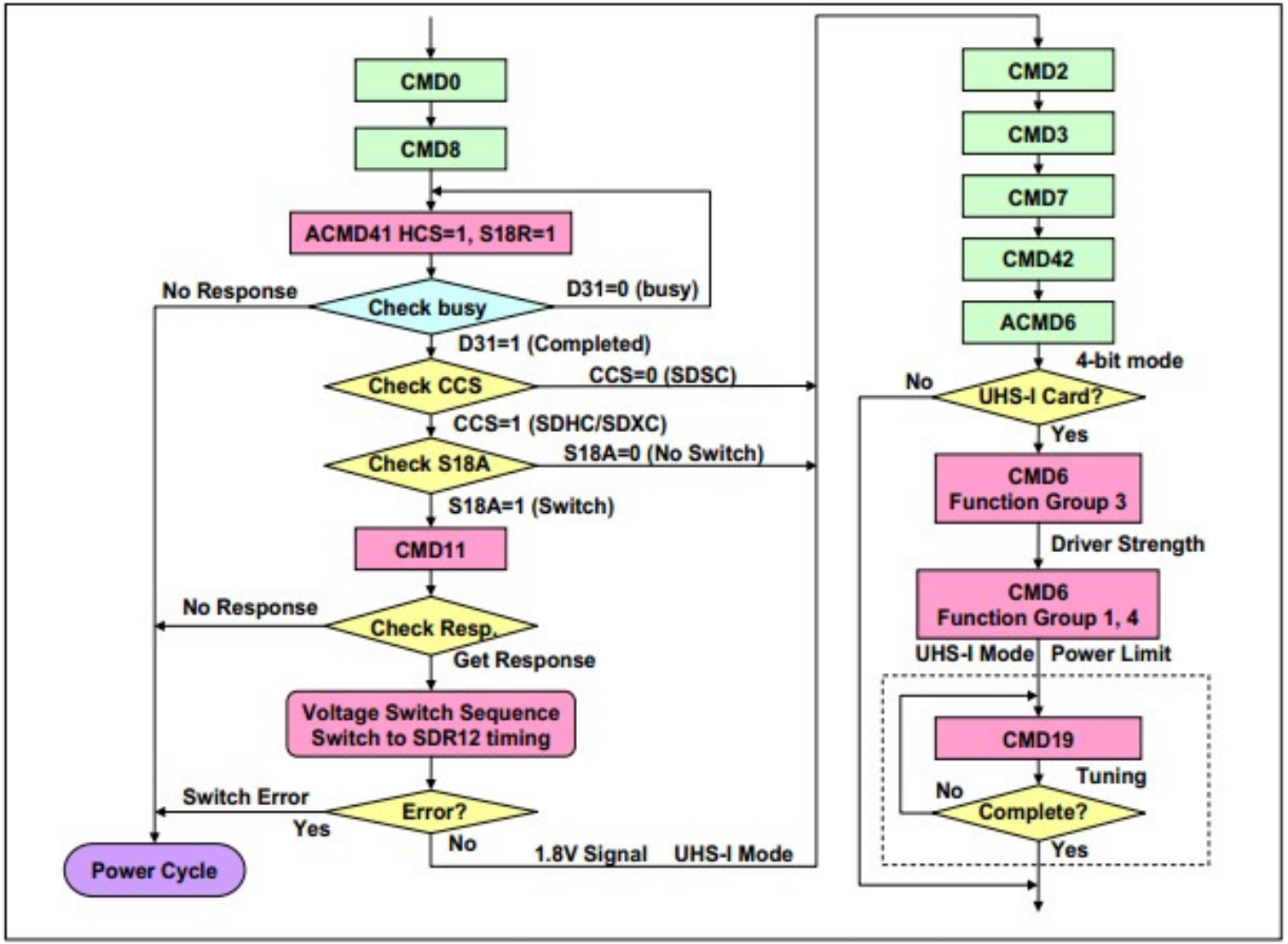
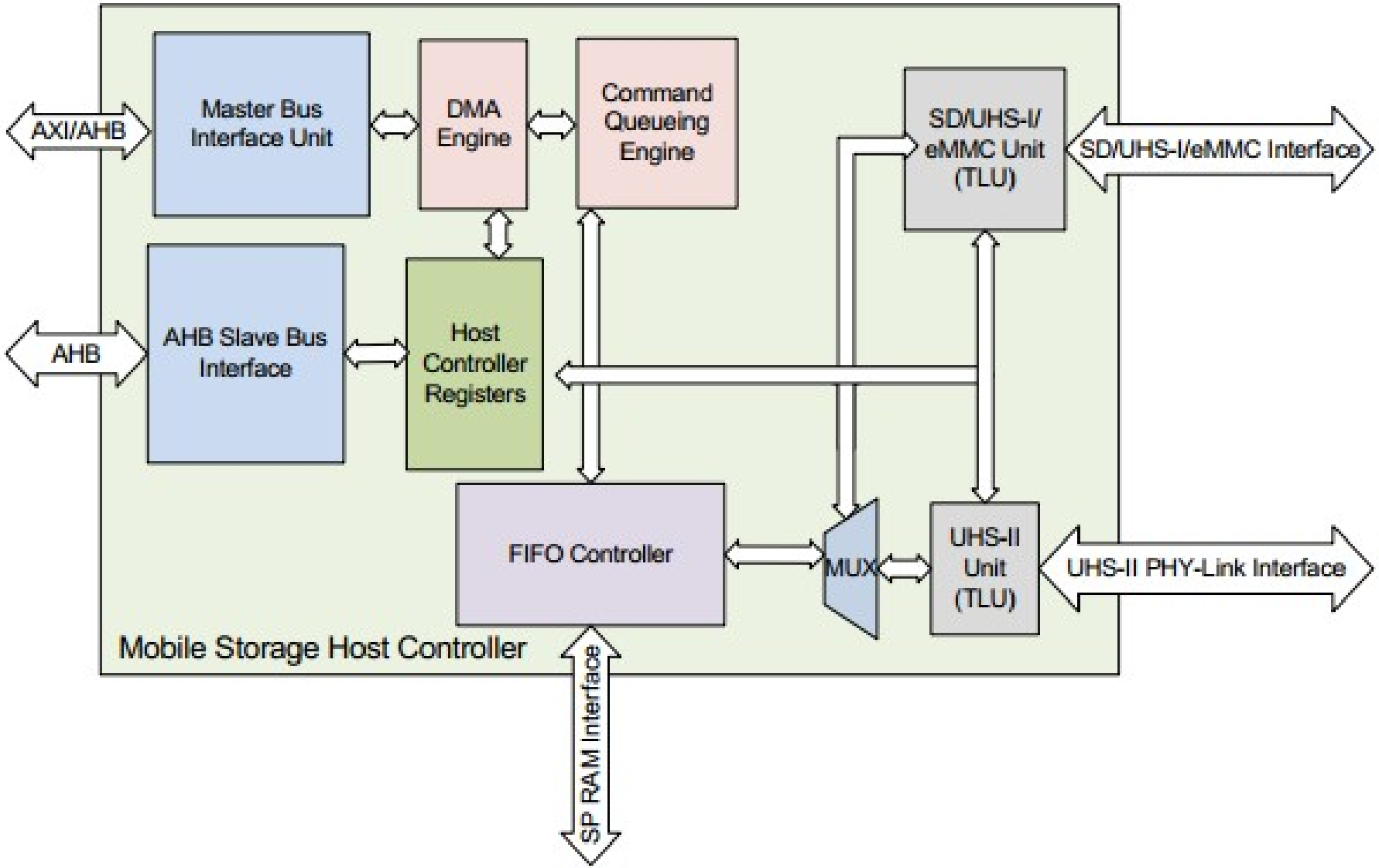
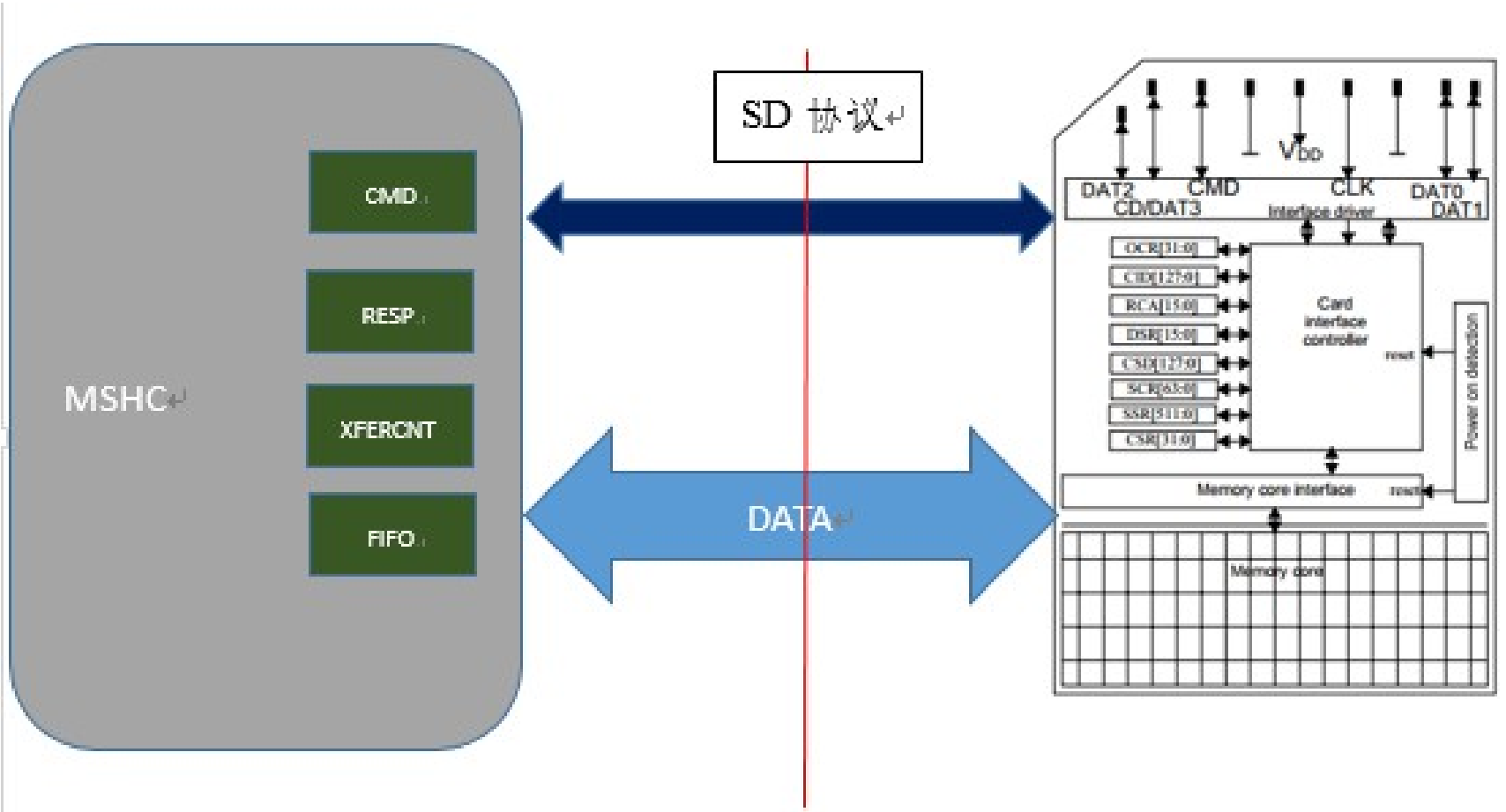


Figure 4-6 : UHS-I Host Initialization Flow Chart

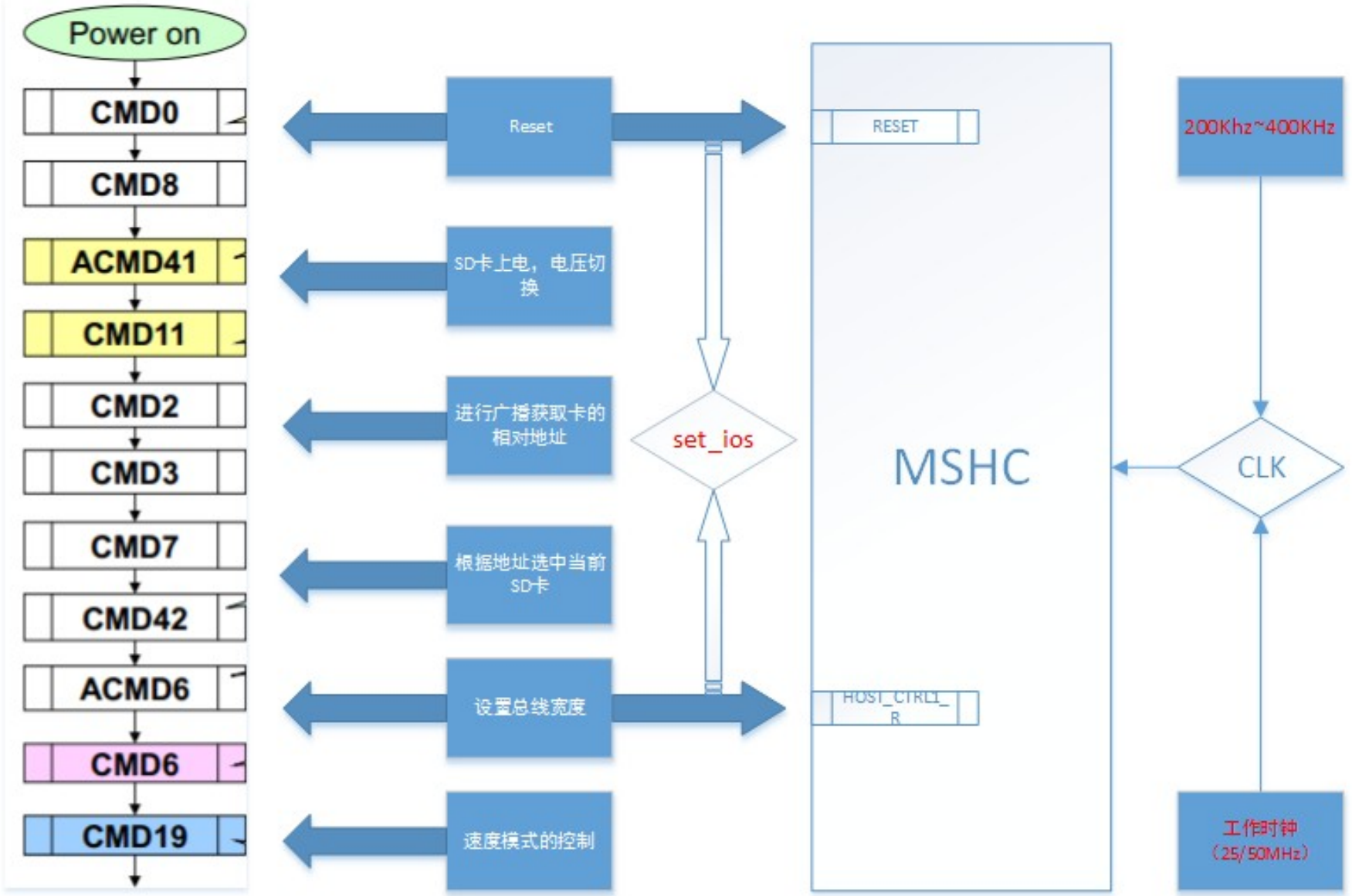
X2000 控制器结构图



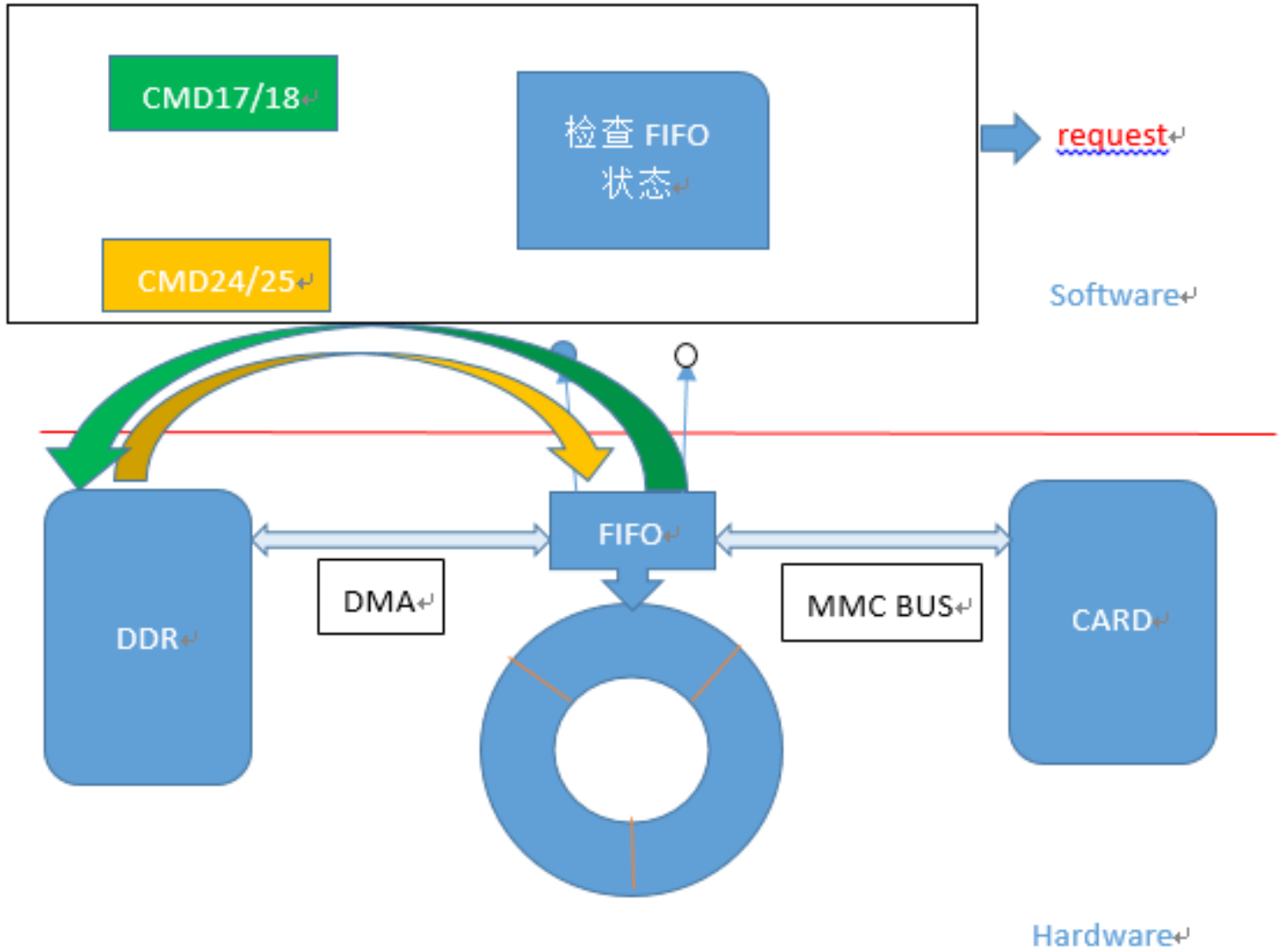
初始化


MSHC
CMD
RESP
XFERCNT
FIFO
SD 协议
DATA
DAT2
CD/DAT3
CMD
CLK
DAT0
DAT1
Card interface controller
Memory core interface
Memory core
Power on detection

初始化流程


Power on
CMD0
CMD8
ACMD41
CMD11
CMD2
CMD3
CMD7
CMD42
ACMD6
CMD6
CMD19
Reset
SD卡上电，电压切换
进行广播获取卡的相对地址
根据地址选中当前SD卡
设置总线宽度
速度模式的控制
set_ios
RESET
MSHC
HOST_CTRL1_R
200Khz~400KHz
CLK
工作时钟（25/50MHz）

数据传输



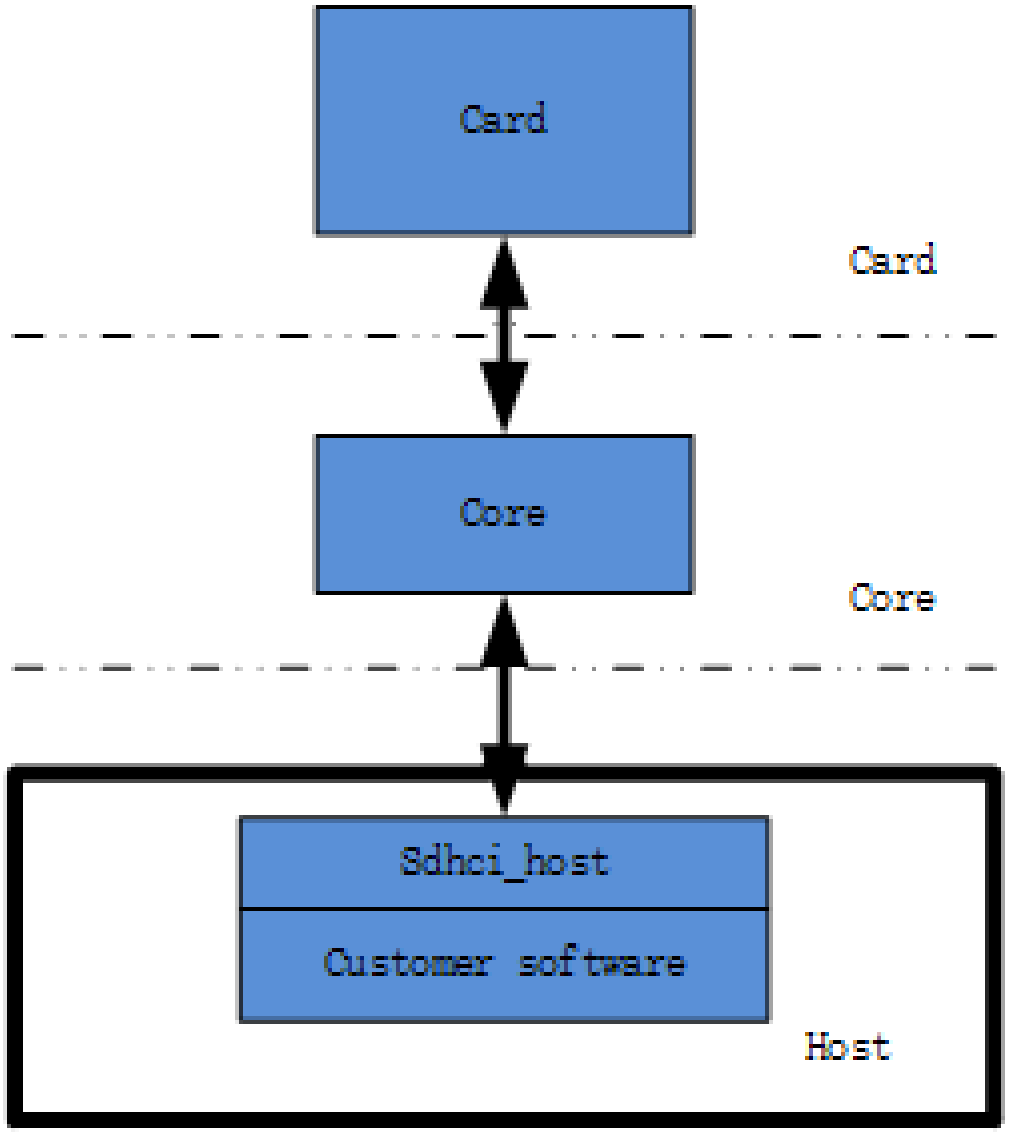
X2000 MSHC

软件实现 ---MMC 子系统

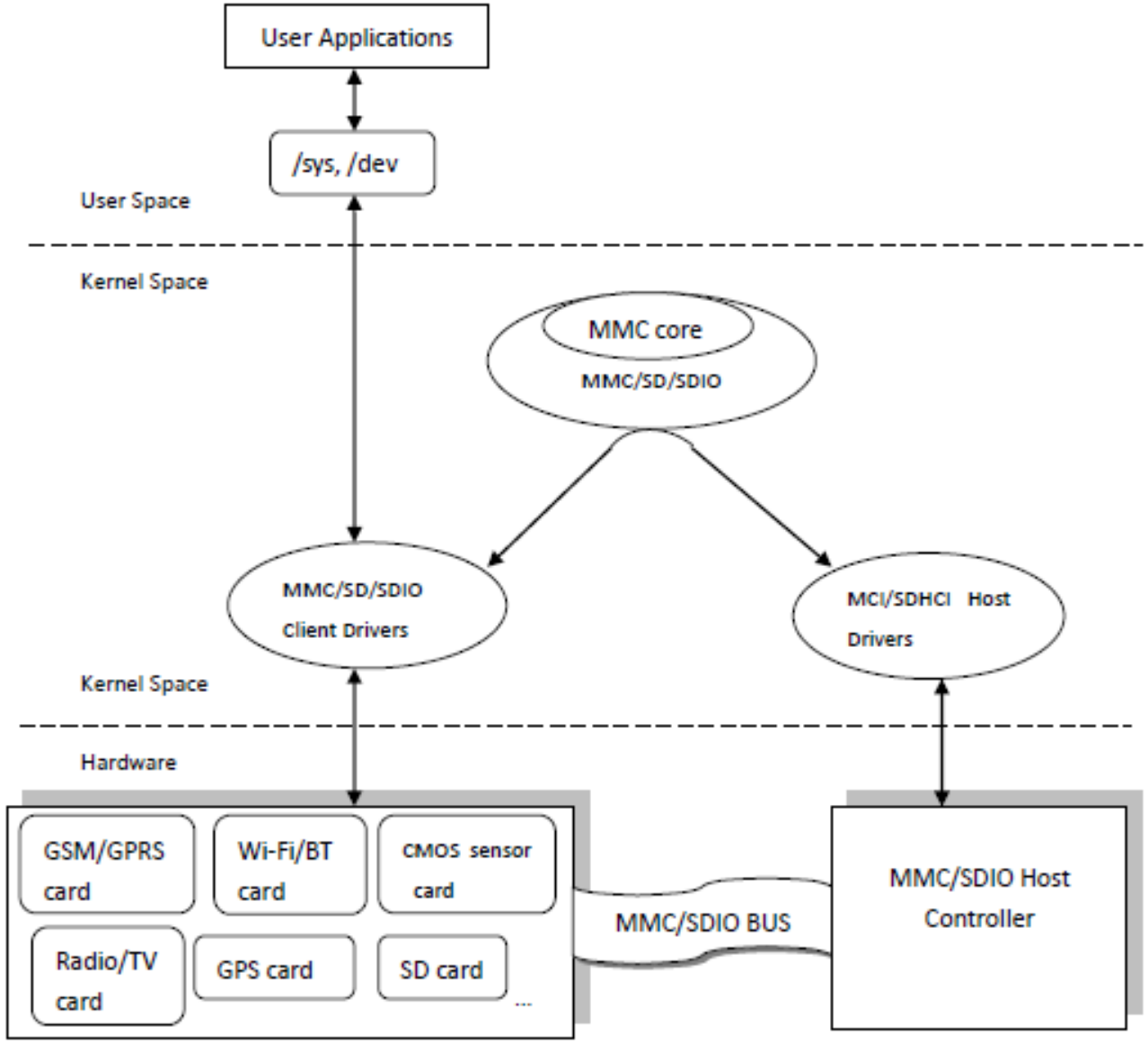
- MMC 子系统代码 drivers/mmc ，共三个目录：

  - Card ：存放闪存卡 (块设备) 的相关驱动
  - Core ：整个 MMC 的核心层，这部分完成不同协议和规范的实现，为 host 层和设备驱动层提供接口函数。
  - Host ：针对不同主机端的 SDHC 、 MMC 控制器的驱动；

X2000 MSHC 驱动软件结构



软件框架


User Applications
/sys, /dev
User Space
Kernel Space
MMC core
MMC/SD/SDIO
MMC/SD/SDIO
Client Drivers
MCI/SDHCI Host
Drivers
Kernel Space
Hardware
GSM/GPRS
card
Wi-Fi/BT
card
CMOS sensor
card
Radio/TV
card
GPS card
SD card
...
MMC/SDIO BUS
MMC/SDIO Host
Controller

数据结构

```
struct mmc_host {
        struct device           *parent; 、
        struct device           class_dev;
        int             index;
        const struct mmc_host_ops *ops;
        struct mmc_ios          ios;     /* current io bus settings */
        struct mmc_card         *card;  /* device attached to this host */
        const struct mmc_bus_ops *bus_ops; /* current bus driver */
        .   .   .

        unsigned long           private[0] ____cacheline_aligned;
};
```

代码： include/linux/mmc/host.h

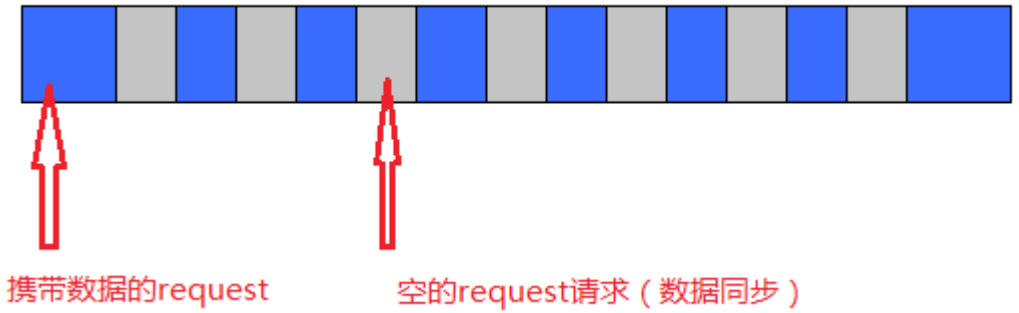
驱动实现

- MMC 驱动的实现主要有两个“线程”
  - 初始化（热插拔）
    - INIT_DELAYED_WORK(&host->detect, mmc_rescan)
  - 数据传输
    - kthread_run(mmc_queue_thread, mq, "mmcqd/%d%s", host->index, subname ? subname : "");
- 三个接口
  - mmc_add_host(struct mmc_host *host)
  - void (*set_ios)(struct mmc_host *host, struct mmc_ios *ios);
  - void (*request)(struct mmc_host *host, struct mmc_request *req);

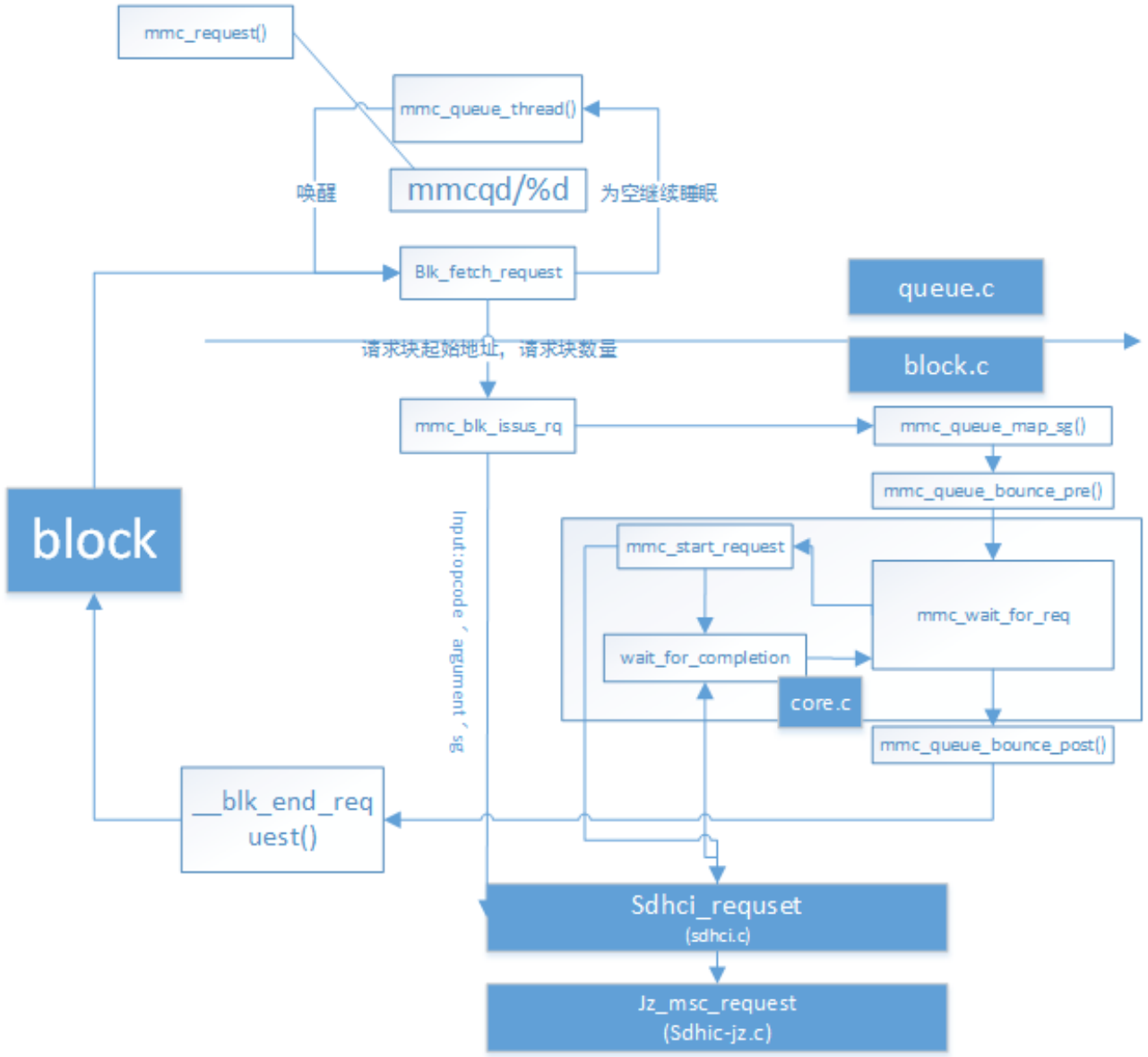
数据传输— request 处理

- Request 请求队列

  Block 将文件系统层传入的数据按照一定的策略进行组合，最后形成一个队列。



数据传输— request 处理



谢谢！