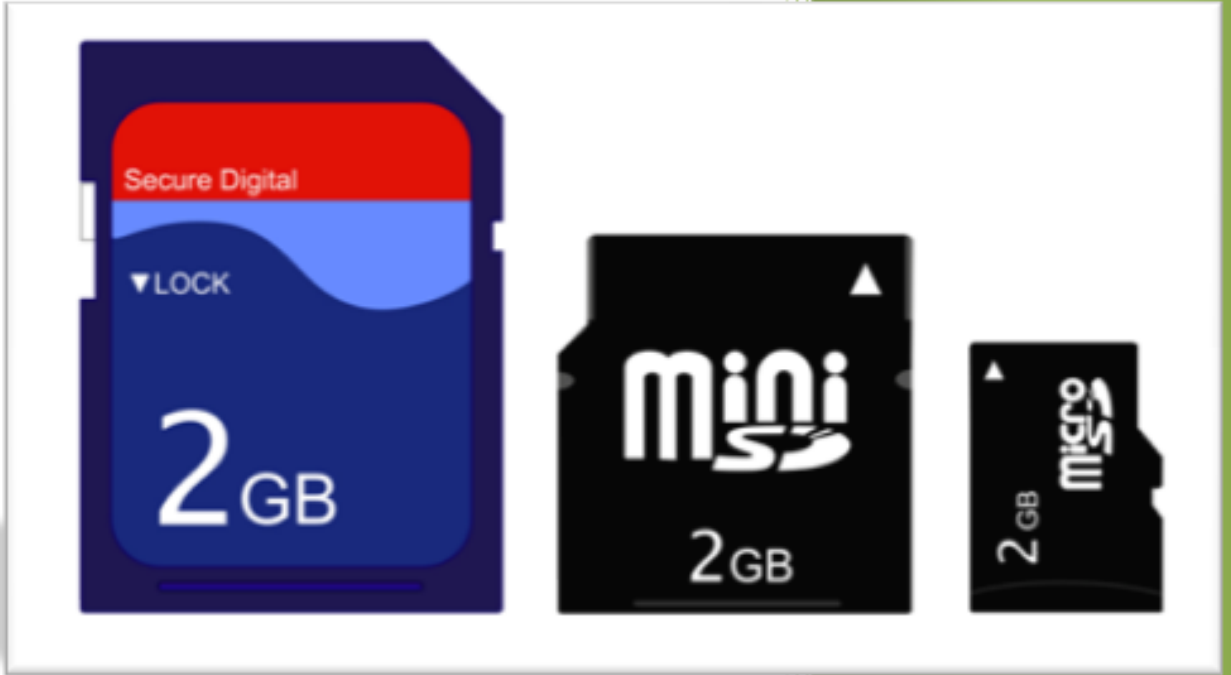


SD Kart Programlama Klavuzu



Bariř Samancı

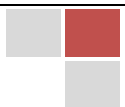
18.08.2010

Önsöz

Bu e-kitapta SD kartlar ve kullanımları üzerine bilgi bulacaksınız. İçeriğinde SD kartın iki haberleşme modu olan SD ve SPI modunun nasıl çalıştığı, veri okuma, veri yazma, silme ve koruma gibi işlemlerin nasıl yapıldığı anlatılmıştır. SD kartların iç yapısı, çalışma biçimleri, SD kart saklayıcıları ve SD kart kontrolcüsü üzerine bilgiler verilmiştir. Kitaptaki bilgiler anlaşıldığı takdirde herhangi bir mikrodenetleyici ile SD kartlara veri yazılıp okunabilir.

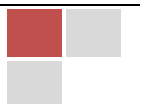
Herkese faydalı olması dileğiyle ...

Barış SAMANCI
Elektronik Mühendisi
www.barissamanci.net
baris@barissamanci.net



İÇİNDEKİLER

Başlık	Sayfa
1. SD Kart Nedir?	4
2. SD Veri Yolu Kullanımı	6
2.1 Tanım	6
2.2 SD Kart Saklayıcıları	9
2.2.1 CID Saklayıcısı	9
2.2.2 RCA Saklayıcısı	10
2.2.3 CSD Saklayıcısı	10
2.2.4 OCR Saklayıcısı	12
2.3 SD Kart Fonksiyonel Çalışma	13
2.4 Kimlik Doğrulama Modu (Evresi)	14
2.4.1 Kart reset	14
2.4.2 Çalışma koşullarını doğrulama:	14
2.4.3 Kart kurulum ve kimlik doğrulama işlemi:	15
2.5 Veri Transfer Modu (Evresi)	17
2.6 Veri yolu genişliği seçimi	19
2.7 Veri Okuma	19
2.7.1 Blok Okuma	19
2.8 Veri Yazma	20
2.8.1 Blok Yazma	20
2.8.2 Çoklu Blok Yazımı Öncesi Ön Silme İşlemi	21
2.8.3 Yazılan Blok Sayısı	22
2.9 Silme	22
2.10 Kart Yazma Koruması	22
2.11 Kartı kilitleme işlemleri	22
2.12 Komutlar ve Cevaplar	23
2.12.1 Komut Formatı	23
2.13 Cevap Formatları	23
3. SPI Veri Yolu Kullanımı	25
3.1 Mod Seçimi ve Kurulum	26
3.2 Veri Yolu Koruması	27
3.3 Veri Okuma	28
3.4 Veri Yazma	29
3.5 Silme ve Yazma Koruma Yönetimi	30
3.6 CID/CSD Saklayıcılarını Okuma	30
3.7 Reset İşlemleri	30
3.8 Cevaplar	30
3.8.1 R1	30
3.8.2 R1b	31
3.8.3 R2	31
3.8.4 R3	32
3.8.5 R7	32
3.9 Kontrol İşaretleri	32
3.9.1 Veri Cevap İşareti (Data Response Token)	32
3.9.2 Blok Başlangıç ve Transferi Durdur İşaretleri	32
3.9.3 Veri Hata İşareti	33
4. Bağlantı	34
4.1 SD bağlantısı	34
4.2 SPI bağlantısı	35

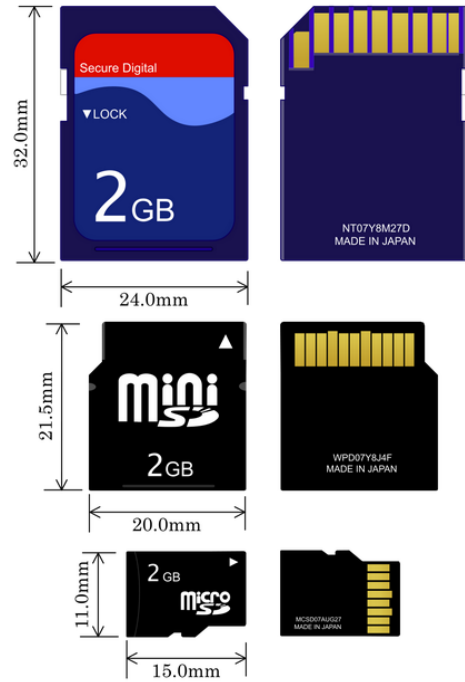


1. SD Kart Nedir?

SD (Secure Digital) kart genelde mobil araçlar ve gömülü sistemler hedeflenerek bu sistemlerin bilgi depolayıp aktarabilmesi için geliştirilmiş flash tabanlı bir hafıza kartıdır. MMC kartın geliştirilmesi ile ilk temelleri atılmıştır. Bu sayede SD ve MMC kartlar uyumlu olarak çalışabilmektedirler. SD kartlar ve bir kaç üretici firmanın desteği ile standarta bağlanmış ve bir topluluk tarafından geliştirilmektedir. Topluluğun web sitesi <http://www.sdcard.org/> dur.

SD kartlar küçük boyutları, veri aktarım hızları ve yüksek boyutları ile en çok tercih edilen hafıza birimlerinin başında gelmektedirler. 2000 yıllarında ilk SD kartlar piyasaya sürüldüğünde 1MB-2MB yazım hızlarına ve 32MB-64MB saklama kapasitelerine sahiptiler. Zamanla birlikte SD kartlar geliştirildi, boyutları dahada küçüldü, yazma hızları ve kapasiteleri arttı. İlk SD kartlar 2GB kapasiteye kadar destek veriyorlardı fakat zamanla SDHC (SD High capacity) kartlar geliştirildi ve 4-8-16-32GB kapasiteleri destekler hale geldi. 2010 yılının başında 2TB destekli SDXC kartlar duyuruldu. Şuanda piyasada 32GB ve 64GB SDXC kartları bulmak mümkün.

SD kartların 3 farklı kılıf tipi vardır. Çalışma biçimleri aynı sadece boyutları, pinleri ve kılıfları farklıdır. Bunlar SD, miniSD ve microSD dir. Yandaki şekilde bu tipleri ve kılıfların boyutlarını görebilirsiniz.



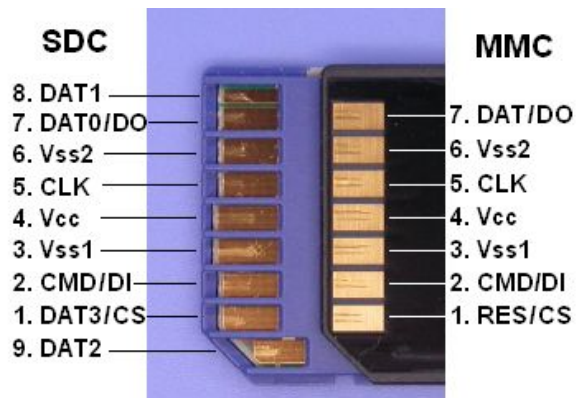
Şekil 1.1 SD card kılıf tipleri ve boyutları



Şekil 1.2 SD kartın iç yapısı.

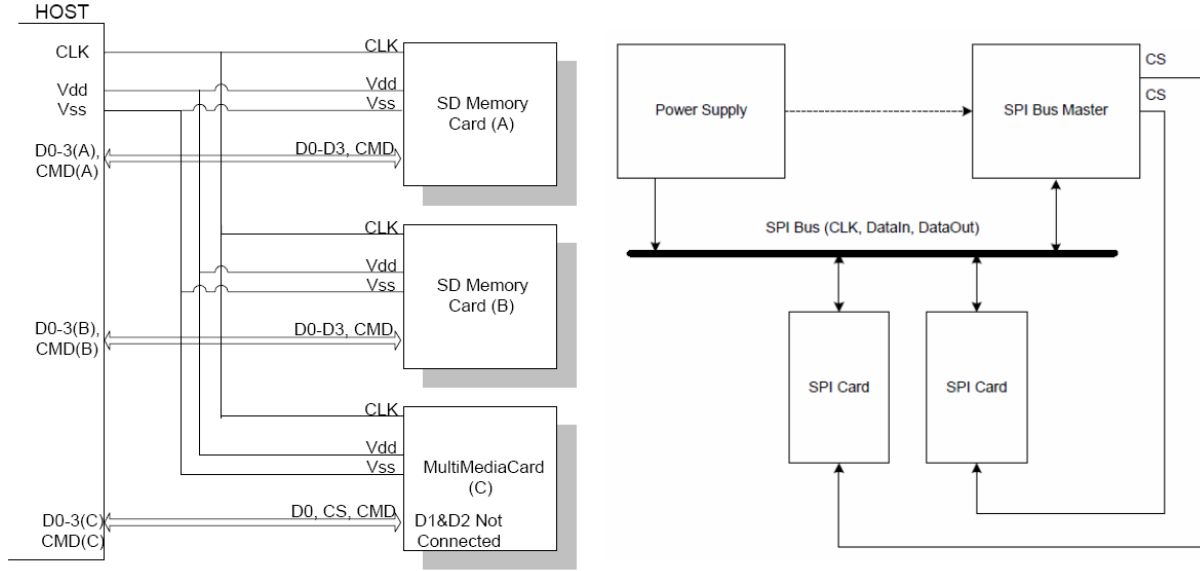
SD kartın içerisinde dahili kontrolcü ve flash hafıza birimi vardır. Kontrolcü beslemesini SD kartın arayüz pinlerinden alır ve bu pinlerden gelen komut ve verileri işleyerek flash hafızaya yazar yada okur. Yandaki şekil 1.2 de panasonic firmasının ürettiği 64MB lık bir SD kartın içerisindeki kontrolcü ve flash hafızayı görebilirsiniz.

SD kartın 9 adet pini vardır. Bunlardan 4 tanesi veri hattıdır ve DAT0-3 ile ifade edilir. 3 tanesi besleme pini, bir tanesi clock pini (CLK), kalan ise (CMD) komut pinidir. Şekil 1.3 te SD kartın ve MMC kartın pin yapısını görebilirsiniz.



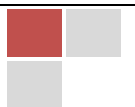
Şekil 1.3 SDC ve MMC kartların pin yapısı

SD kartların iki adet çalışma modu vardır. Bu modlar SD modu (native – varsayılan) ve SPI modudur. Kartın varsayılan haberleşme modu SD modudur. Bu modda veri DAT0-4 hatlarından 4 bitlik paralel veriler bitler şeklinde akmaktadır. SPI modunda ise data-in ve data-out pinlerinden haberleşilmektedir. SPI modunun avantajı, SPI donanımı olan bir çok mikroişlemci ile haberleşme sağlanabilmesidir. SD modunun avantajı ise SPI moduna göre çok daha hızlı bir şekilde okuma ve yazma yapılabilmesidir. Şekil 1.4 te SD ve SPI topolojilerini görebilirsiniz.



Şekil 1.4 SD veri yolu (soldaki) ve SPI veri yolu (sağdaki) topolojileri

Bu dökümanda önce SD, sonra SPI modu anlatılacaktır. SPI modu okunmadan önce SD modu okunursa daha faydalı olacaktır. Anlatımlar minimize edilerek yapılacaktır. Komut sınıf tabloları, komut tabloları anlatılmayacak, CSD saklayıcısı ve bazı saklayıcılar sadeleştirilecektir. Daha detaylı bilgi isterseniz <http://www.sdcard.org/developers/tech/sdcard/pls/> adresinden kullandığım kaynakları indirip bu tablolara göz atabilirsiniz.



2. SD Veri Yolu Kullanımı

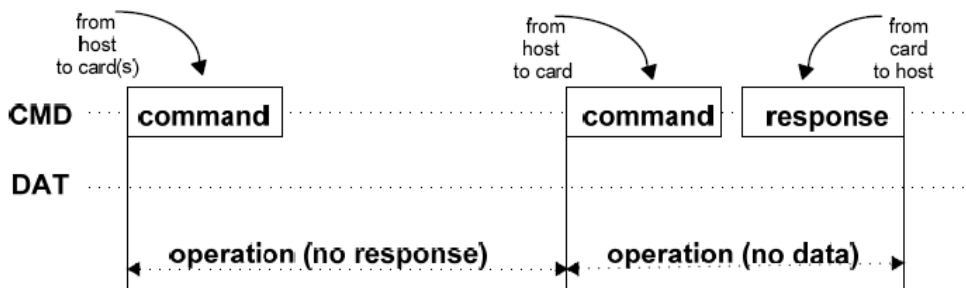
2.1 Tanım

SD veri yolu üzerindeki haberleşmeler start biti ile başlatan ve stop biti ile biten komut, cevap ve veri iletimi üzerine kuruludur.

Komut: Komutlar istenilen işlemleri başlatırlar veri okuma veya yazmamıza yardımcı olurlar. Komutlar işlemciden tek bir karta (adreslenmiş komut) veya sisteme bağlı olan tüm kartlara (broadcast komut) gönderilebilir. Komut hattı (CMD Line) bir bitliktir ve komutlar seri olarak iletilir.

Response: Cevaplar komutu alan kart tarafından veya kartlar tarafından komut okunduktan sonra işlemciye gönderilir. Bazı komutlara cevap gönderilirken bazılarına gönderilmez. Cevaplarda CMD komut hattı üzerinden seri şekilde iletilirler.

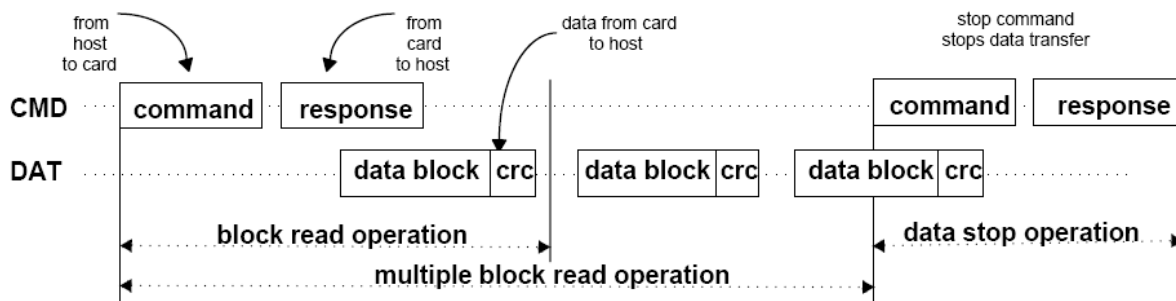
Veri: Veriler işlemciden karta veya karttan işlemciye şeklinde aktarılabilirler. Veri aktarımı için veri hatları (DAT) kullanılır.



Şekil 2.1 Veri yolu üzerindeki komut ve cevap paketleri

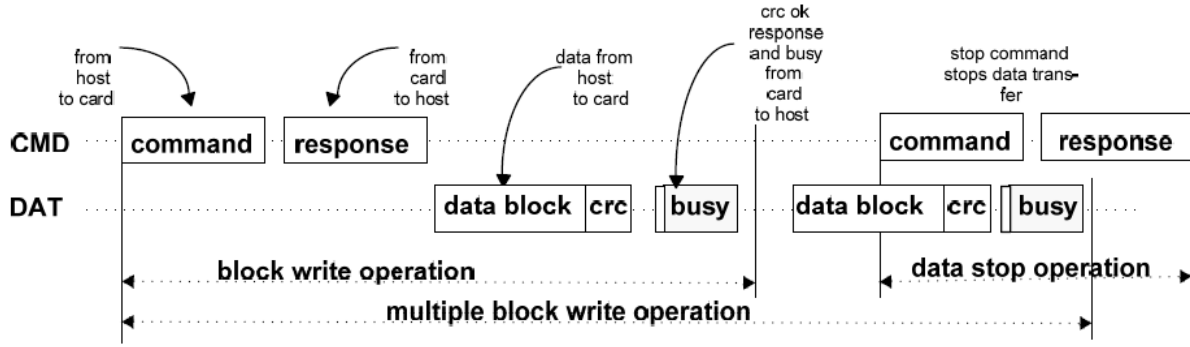
En basit haliyle komut gönderme veya gönderilen komuta karşılık düşen cevabı okuma yukarıdaki şekilde gibidir. Yukarıdaki şekilde komut ve cevaplar veri taşıyamaktadır fakat bazı komut ve bu komutlara verilen cevaplar veri taşıyabilirler.

Veri okuma ve yazma işlemleri bloklar halinde yapılmaktadır. Veriler tek veya çoklu bloklar halinde taşınabilirler ve CRC ile hata kontrolleri yapılır. Çoklu bloklar kullanılırsa büyük verilerin okunması ve yazılmasında hız kazanılmaktadır. Aşağıdaki şekilde veri okuma yöntemleri gösterilmektedir.



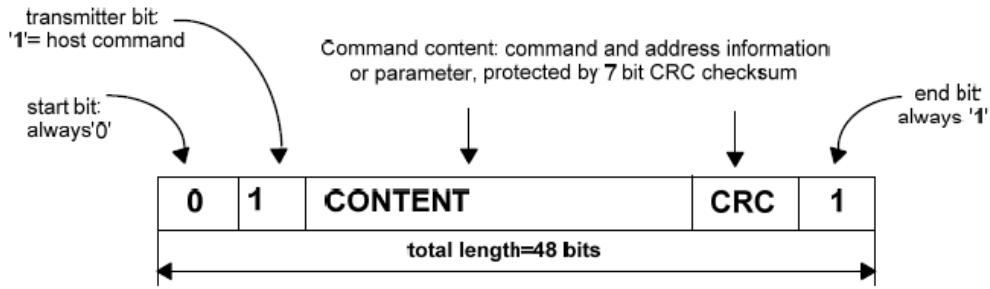
Şekil 2.2 Blok okuma işlemleri

Veri yazma işlemleri ise biraz farklıdır. Veri yazarken kart işlemcisi DAT0 hattını kullanarak meşgul olduğunu belirtir. Bir sonraki veri bloğu kart işlemcisi yazma işini tamamladıktan sonra gönderilmelidir. Şekil 2.3 te gösterilmiştir.



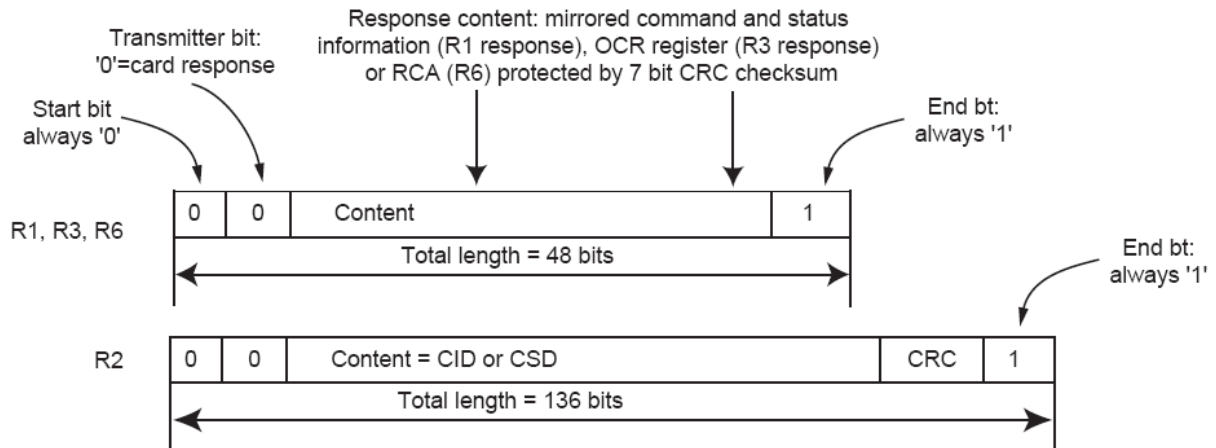
Şekil 2.3 Blok yazma işlemleri

Komutların çerçeve formatı aşağıdaki gibidir.



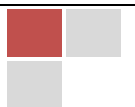
Şekil 2.4 Komutların 48bit uzunluğundaki temel çerçeve yapısı

Komutlar 48 bit uzunluğundadır ve başlangıç biti her zaman 0, bitiş biti ise her zaman 1 dir. İçerik kısmında komut kodu, adres bilgisi ve parametre taşınabilir. Bu bilgiler yedi bitlik CRC ile korunmaktadır. Cevap formatı ise aşağıdaki şekillerdeki gibi olabilir. R1, R3, R6 formatları 48 bitlik CID bilgisini veya CSD bilgisini taşıyan cevap çerçevesi 136 bit uzunluğundadır.



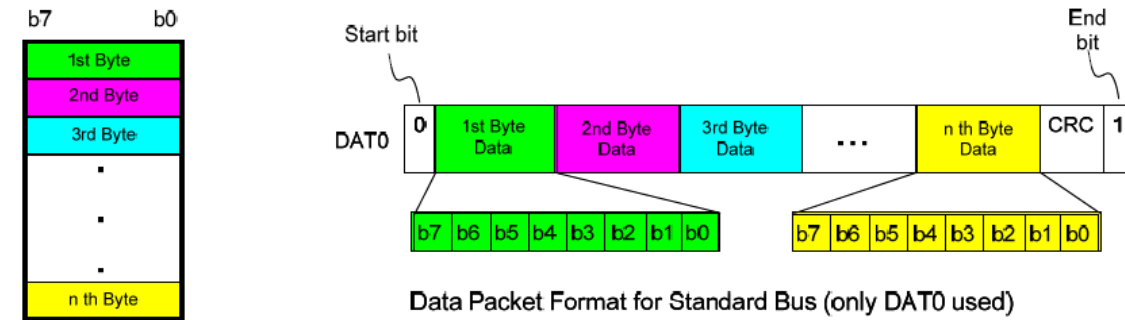
Şekil 2.5 Cevap tipleri ve temel çerçeve yapıları

CMD hattında en değerli bit önce, en değersiz sonra gönderilmektedir. Geniş veri yolu seçeneği seçildiğinde veriler DAT0-DAT3 hatlarından iletilmektedirler. Bu iletişimde her hat için ayrı ayrı start, stop biti kullanılır ve CRC bilgileri her hat için ayrı ayrı olarak hesaplanır. CRC durum bilgisi ve meşgul bilgisi karttan işlemciye sadece DAT0 hattı ile iletilmektedir. Bu esnada DAT1-DAT3 hatlarındaki bilgi önemsizdir.



SD kartlarda iki adet veri paketi formatı vardır. Bunlar

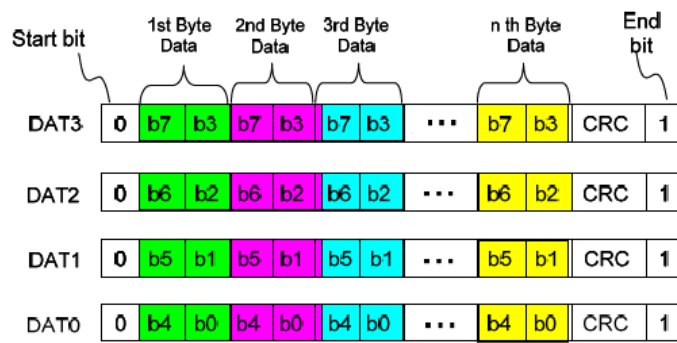
1. **Genel veri formatı:** Bu format 8 bitliktir. Değerli byte en sonda, değersiz byte başta gönderilir. Her byte'ın bitlerinde ise değersiz bit sonda, değerli bit başta gönderilir. Şekil 2.6 da anlatılanlar gösterilmiştir.
2. **Geniş veri formatı:** Bu formatta verinin bitleri en değerli bitten başlanarak gönderilir. Gönderim şekli Şekil 2.7 de gösterilmiştir.



8bit width Data

Ex

[SDIO]
CMD53
[SD memory]
CMD17, CMD18,
CMD24, CMD25,
ACMD18, ACMD25,
etc



Data Packet Format for Wide Bus (all four lines used)

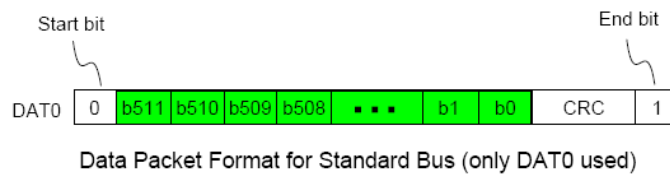
Şekil 2.6 Sekiz bitlik genel veri formatı



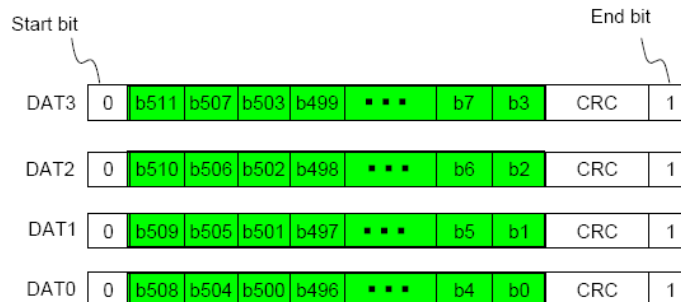
Wide Width
Data

Ex.

[SD memory]
ACMD13(SD Status),
ACMD51(SCR),
etc



Data Packet Format for Standard Bus (only DAT0 used)



Data Packet Format for Wide Bus (all four lines used)

Şekil 2.7 Geniş veri formatı

2.2 SD Kart Saklayıcıları

SD kartlarda bir takım saklayıcılar vardır. Bu saklayıcılar bize kartın üretimi, durumu ve çalışması esnasında bir takım bilgiler sağlarlar. Bu saklayıcılar tablo 2.1 de gösterildiği gibidirler.

İSİM	UZUNLUK	AÇIKLAMA
CID	128	Kartın kimlik doğrulama için bireysel ID numarası. (Zorunlu)
RCA	16	Karta kurulum esnasında işlemci tarafından atanan kart adresi (Zorunlu)
DSR	16	İsteğe bağlı olarak veri yolunun performansını arttırmak için genişletilmiş çalışma koşullarını kullanmamızı sağlar. Varsayılan olarak 0x404 tür (Seçmeli)
CSD	128	(Card Specific Data) Bu saklayıcı karta özel çalışma şartlarını saklar. Ayrıntılı bilgi için SD kart datasheetine bakabilirsiniz. (Zorunlu)
SCR	64	SD Konfigürasyon saklayıcısı. SD kartın özellikleri yetenekleri hakkında bilgi sağlar. Ayrıntılı bilgi için SD kart datasheetine bakabilirsiniz. (Zorunlu)
OCR	32	(Operation conditions register) Kartın V_{DD} besleme gerilimi ve power up zamanı için bilgi bitlerini içerir. (Zorunlu)
SSR	512	SD Status Register. Kartın tescilli özellikleri hakkında bilgi verir. (Zorunlu)
CSR	32	Card Status Register. Kartın durumu hakkında bilgi verir. (Zorunlu)

Tablo 2.1 SD kart saklayıcıları

2.2.1 CID Saklayıcısı

Kartın kimlik ve üretim bilgilerini saklar. 128 bit genişliğindedir. Her kartın benzersiz bir CID saklayıcısı vardır. Alanları tablo 2.2 de gösterilmiştir.

İSİM	ALAN	GENİŞLİK	CID ARALIĞI
Üretici ID	MID	8	[127:120]
OEM/Uygulama ID	OID	16	[119:104]
Ürün adı	PNM	40	[103:64]
Ürün revizyonu	PRV	8	[63:56]
Ürün seri numarası	PSN	32	[55:24]
Rezerve	-	4	[23:20]
Üretim tarihi	MDT	12	[19:8]
CRC7	CRC	7	[7:1]
Rezerve (her zaman 1)	-	1	[0:0]

Tablo 2.2 CID saklayıcısı

MID:

SD kartın 8 bitlik üretici ID sidir. Üretici ID leri SD-3C, LLC tarafından belirlenir.

OID:

Kartın 2 karakterlik ASCII dizisidir. SD-3C, LLC tarafından belirlenir.

PNM:

5 karakterlik ASCII dizisidir. Ürün adı bilgisini saklar.

PRV:

Ürünün revizyon numarasını saklar 8 bit genişliğindedir. Üst dört bit tam sayı kısmını, alt dört bit ondalık kısmı saklar. Örneğin 6.2 0110 0010b dir

PSN:

Kartın 32 bitlik seri numarasını saklar.



MDT:

Kartın üretim tarihini saklar. 12 bitliktir. Üst 8 bit [19:12] üretilen yılı, alt 4 bit [11:8] üretim ayını saklar. Üretim yılı 2000 den itibaren sayılır. Örneğin Nisan 2001, 00000001 0100b olacaktır.

2.2.2 RCA Saklayıcısı

16 bitlik kart adresini saklar. Kartın kurulumu esnasında kimlik doğrulama aşamasında işlemci tarafından okunur. İşlemci RCA adresini kullanarak hangi karta yazacağını veya okuyacağını kartlara bildirir. Eğer sistemde birden fazla kart varsa sadece RCA adresi belirtilen kart işlem yapar. Aynı zamanda RCA adresi 0x0000 geçilerek kart seçimi yapılırsa sistemdeki tüm kartlar bekleme (standby) durumuna geçerler.

2.2.3 CSD Saklayıcısı

128 bitlik bir saklayıcıdır. Bu saklayıcı kartın hafızası, haberleşme hızı, sınıfı gibi bilgileri içermektedir. İki farklı versiyonu vardır ve kartın tipine göre bu yapılardan biri kullanılmaktadır. Versiyon numarası [127:126] bitlerinde saklıdır. Tablo 2.3 te gösterilmiştir.

BIT	Yapı versiyonu	Versiyon / Kart Kapasitesi
0	CSD versiyon 1.0	Versiyon 1.01 – 1.10 Versiyon 2.0 / Standart kapasite (SD)
1	CSD versiyon 2.0	Versiyon 2.00 / SDHC (High Capacity) Versiyon 2.00 / SDXC (Extended Capacity)
2-3	rezerve	-

Tablo 2.3 CSD saklayıcısı versiyon alanı

Yukarıdaki tablodan kullanılan kartın hangi CSD yapısına sahip olduğu anlaşılır. Görüldüğü gibi iki tablo vardır ve 128 bit uzunluğundadırlar. Bu yüzden tabloları burada önemli gördüğüm alanlara sadeleştirerek anlatacağım. Detaylı bilgi için sd kart datasheetlerine bakabilirsiniz.

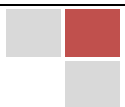
CSD Register (CSD Version 1.0):

İSİM	ALAN	GENİŞLİK	R/W	ARALIK
CSD Yapısı	CSD_STRUCTURE	2	R	[127:126]
Kart Komut Sınıfı	CCC	12	R	[95:84]
Max. veri bloğu uzunluğu (Okuma)	READ_BL_LEN	4	R	[83:80]
Kart boyutu	C_SIZE	12	R	[73:62]
Kart boyutu çarpanı	C_SIZE_MULT	3	R	[49:47]
erase sector size	SECTOR_SIZE	7	R	[45:39]
Max. veri bloğu uzunluğu (Yazma)	WRITE_BL_LEN	4	R	[25:22]
Geçişi yazma koruması	TMP_WRITE_PROTECT	1	R/W	[12:12]

Tablo 2.4 CSD v1.0 saklayıcısının sadeleştirdiğim yapısı

CCC:

CCC alanı bize kartın hangi komut sınıflarını desteklediğini söyler. 12 bitlik bit alandır ve 1 olan bitler o komut sınıfının desteklediği anlamına gelir. Komut sınıfı tabloları için datasheete bakabilirsiniz.



CCC Bit	Desteklenen komut sınıfı
0	Sınıf 0
1	Sınıf 1
...	...
11	Sınıf 11

Tablo 2.5 CSD saklayıcısındaki CCC alanı

READ_BL_LEN:

Okunacak veri bloğunun max. boyunu bildirir. Bu boyut 512-2048 byte aralığındadır ve $2^{\text{READ_BL_LEN}}$ ile hesaplanır. SD kartta her zaman WRITE_BL_LEN değeri ile READ_BL_LEN değeri birbirine eşittir.

READ_BL_LEN	Blok Uzunluğu
0-8	rezerve
9	$2^9 = 512$ byte
10	$2^{10} = 1024$ byte
11	$2^{11} = 2048$ byte
12-15	rezerve

Tablo 2.6 CSD saklayıcısındaki READ_BL_LEN alanı

C_SIZE:

Bu parametre kullanıcının kullanabileceği kart kapasitesinin hesaplanmasında kullanılır. Hafıza kapasitesi şu bilgilerin kullanımı ile hesaplanır, C_SIZE, C_SIZE_MULT, READ_BL_LEN.

$$\text{Hafıza kapasitesi} = \text{BLOCKNR} * \text{BLOCK_LEN}$$

$$\text{BLOCKNR} = (\text{C_SIZE} + 1) * \text{MULT}$$

$$\text{MULT} = 2^{\text{C_SIZE_MULT} + 2} \quad (\text{C_SIZE_MULT} < 8)$$

$$\text{BLOCK_LEN} = 2^{\text{READ_BL_LEN}}, \quad (\text{READ_BL_LEN} < 12)$$

C_SIZE_MULT:

Kart kapasitesinin hesaplanmasında çarpan görevi üstlenir.

C_SIZE_MULT	MULT	C_SIZE_MULT	MULT
0	$2^2 = 4$	4	$2^6 = 64$
1	$2^3 = 8$	5	$2^7 = 128$
2	$2^4 = 16$	6	$2^8 = 256$
3	$2^5 = 32$	7	$2^9 = 512$

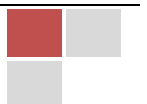
Tablo 2.7 CSD saklayıcısındaki C_SIZE_MULT alanı

SECTOR_SIZE:

Silinebilir sektör boyutunu bildirir. 7 bitlik ikilik sistemde kodlanmıştır. Gerçek değeri bir fazlasıdır. Yani sıfır bir, 127 128 anlamına gelir. WRITE_BL_LEN ile ilişkilidir.

WRITE_BL_LEN:

Yazılacak veri bloğunun max. boyunu bildirir. Bu boyut 512-2048 byte aralığındadır ve $2^{\text{WRITE_BL_LEN}}$ ile hesaplanır. SD kartta her zaman WRITE_BL_LEN değeri ile READ_BL_LEN değeri birbirine eşittir.



WRITE_BL_LEN	Blok Uzunluğu
0-8	rezerve
9	2 ⁹ = 512 byte
10	2 ¹⁰ = 1024 byte
11	2 ¹¹ = 2048 byte
12-15	rezerve

Tablo 2.8 CSD saklayıcısındaki WRITE_BL_LEN alanı

TMP_WRITE_PROTECT:

Geçici yazma korumasıdır. Bu bit hem yazılabilir hem okunabilir. Karta veri yazma, silme gibi işlemleri geçici olarak engeller. Varsayılan değeri 0 yani koruma yoktur.

CSD Register (CSD Version 2.0)

CSD versiyon 2 de gelişen SD kart teknolojisi ile bir takım değişiklikler yapılmıştır. Bizim tablomuzda sadece CSIZE değişmiştir, C_SIZE_MULT kalkmıştır. Daha ayrıntılı bilgi için datasheete bakınız.

İSİM	ALAN	GENİŞLİK	R/W	ARALIK
CSD Yapısı	CSD_STRUCTURE	2	R	[127:126]
Kart Komut Sınıfı	CCC	12	R	[95:84]
Max. veri bloğu uzunluğu (Okuma)	READ_BL_LEN	4	R	[83:80]
Kart boyutu	C_SIZE	22	R	[69:48]
erase sector size	SECTOR_SIZE	7	R	[45:39]
Max. veri bloğu uzunluğu (Yazma)	WRITE_BL_LEN	4	R	[25:22]
Geçici yazma koruması	TMP_WRITE_PROTECT	1	R/W	[12:12]

Tablo 2.9 CSD v2.0 saklayıcısının sadeleştirdiğim yapısı

C_SIZE:

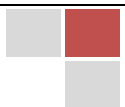
Bu alan 22 bite genişletilmiştir. 2TB a kadar bilgi saklanabilmektedir. Değişen bağıntı aşağıdaki gibidir.

$$\text{memory capacity} = (\text{C_SIZE}+1) * 512\text{K byte}$$

2.2.4 OCR Saklayıcısı

32 bitlik OCR saklayıcısı kartın çalışma koşullarını ve kapasitesini belirleyen bitleri saklar. Ayrıca bit7 ile kartın çift voltaj çalışma aralığına sahip olup olmadığı test edilebilir. Varsayılan olarak yedinci bit 0 dir. Eğer çift voltaj aralığında çalışma özelliğine sahip bir kart CMD8 komutunu alırsa bu biti set eder, eğer çift voltaj aralığında çalışma özelliği yoksa CMD8 den sonra bu bit 0 dir.

Kartın hangi voltajlarda çalışabileceği bu saklayıcıda tutulmaktadır. [23:0] aralığında kartın çalışabileceği voltaj aralığı bildirilmiştir. Ayrıca 31. bit kart voltajının çalışma aralığına kadar yükselip yükselmediğini bildiren power up bitidir. Eğer kart power up rutinini tamamlamadıysa bu bit 0 dir. 30. bit bize kartın hafıza kapasitesi hakkında bilgi verir. Bu bit 31. bit 1 olduktan sonra test edildiğinde 1 okunuyorsa kart SDHC (High Capacity) kart olduğunu söyler. 0 ise standart SD kart olduğunu bildirir. Bu biti test ederek kartın SD kartını yoksa SDHC kartını olduğunu anlayabiliriz.



OCR Bit	Açıklama	OCR Bit	Açıklama
0-3	Rezerve	15	2.7-2.8
4	Rezerve	16	2.8-2.9
5	Rezerve	17	2.9-3.0
6	Rezerve	18	3.0-3.1
7	Düşük voltaj aralığı için rezerve	19	3.1-3.2
8	Rezerve	20	3.2-3.3
9	Rezerve	21	3.3-3.4
10	Rezerve	22	3.4-3.5
11	Rezerve	23	3.5-3.6
12	Rezerve	24-29	24-29
13	Rezerve	30	Kart Kapasite Durumu (CCR)
14	Rezerve	31	Kart power up durumu (busy)

Tablo 2.10 OCR saklayıcısı

2.3 SD Kart Fonksiyonel Çalışma

Kart ve işlemci arasındaki tüm trafiği işlemci yönetmektedir. İşlemci iki tür komut kullanmaktadır. Bunlar broadcast (yayın) ve adreslenmiş (uçtan uca) komutlardır.

Broadcast komutlar:

Broadcast komutlar tüm kartlara yönelik olarak gönderilirler. Bu komutlardan bazılarında cevap verilir.

Adreslenmiş (uçtan-uca komutlar):

Bu komutlar sadece adreslenmiş karta gönderilirler. Kart komutları cevaplar. Tüm SD kart sistemleri için iki adet mod tanımlanmıştır. Bu modlar hem işlemci hem kart için geçerlidir.

Kart kimlik doğrulama modu:

İşlemci resetten sonra bu moddadır ve sisteme bağlı kartları tarar. Kartlar ise resetten sonra bu moda geçerler ve SEND_RCA (CMD3) komutu gelene kadar kimlik doğrulama modunda kalırlar.

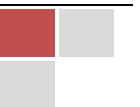
Veri transfer modu:

Kartlar CMD3 komutu geldikten sonra veri transfer moduna geçerler. İşlemci veri yolundaki tüm kartları taradıktan sonra veri transfer moduna geçmelidir.

Tablo 2.11 de modlar ve modlara karşılık düşen kart durumları verilmiştir.

Kart Durumu	İşlem Modu (Evre)
Pasif (inactive)	Pasif (inactive)
Boşta durumu (Idle)	Kart kimlik doğrulama modu
Hazır durumu (Ready)	
Kimlik doğrulama durumu (Identification)	Veri transfer modu
Bekleme modu (Stand by)	
Transfer modu (Tran State)	
Veri gönderme durumu (Sending Data)	
Veri alma durumu (Receive Data)	
Programlama durumu (Programming)	
Bağlantı sonlandırma modu	

Tablo 2.11 Kartın içerisinde bulunabileceği evreler ve durumlar.



2.4 Kimlik Doğrulama Modu (Evresi)

Kimlik doğrulama modunda işlemci tüm kartları resetler. Çalışma şartlarını (voltaj aralığı gibi) belirleyip kartları tanımlar ve RCA adreslerini göndermelerini ister. RCA adresini gönderen kart transfer moduna geçmiş ve bekleme durumundadır. Kimlik doğrulama modunda tüm haberleşme CMD hattından yapılır. Bu modda max çalışma frekansı 400Khz olmalıdır.

2.4.1 Kart reset

Kartlara enerji verildikten sonra kartlar boşa (IDLE) durumuna geçerler. Bu durumdan önce pasif modundadırlar ve hiç bir şekilde tepki vermezler. Eğer işlemci kartları resetlemek istiyorsa kartlara CMD0 komutunu göndermelidir. Bu komutu alan kart yazılımsal olarak resetlenir ve boşa durumuna geri döner.

2.4.2 Çalışma koşullarını doğrulama

Haberleşmeye başlamadan önce kart ve işlemci birbirlerinin çalışma gerilimlerinden haberdar değillerdir. Haberleşebilmek için çalışma gerilimlerinin uyuşması gerekmektedir. İşlemci çalışma gerilimini CMD0 komutu gönderdikten sonra CMD8 komutu göndererek karta bildirir. Kart CMD8 komutuna cevap vererek kendi çalışma gerilimini işlemciye haber verir. İşlemci karta CMD8 SEND_IF_COND gönderdikten sonra kart işlemcinin gönderdiği çalışma gerilimini kontrol eder. Eğer destekliyorsa işlemciye yanıt verir. Desteklemiyorsa cevap vermeden Idle durumuna geri döner.

Çalışma gerilimi CMD8 in parametre alanındaki VHS alanı ile belirlenir. Bu alandaki bitlerden aynı anda sadece bir tanesi 1 olabilir. Hata doğrulama CRC bitleri ve CMD8 in parametre alanındaki kontrol şablonu ile sağlanabilir. Eğer kart işlemcinin gönderdiği çalışma aralığını kabul etmiş ise kontrol şablonu ve çalışma gerilimini cevap olarak geri döner.

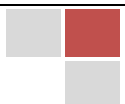
CMD8 komutu "Physical Layer Specification Version 2.00" sürümü ile komut setine eklenmiştir. V2.00 öncesi kartlar bu komutu tanımazlar ve cevaplamazlar. Buradanda voltaj kontrolünün yanında kartın v1.X yada V2.0 ve üstü bir kart olduğunu anlayabiliriz. Aşağıdaki tabloda CMD8 in içeriği gösterilmiştir. Kontrol şablonu alanının 10101010 geçilmesi tavsiye edilmektedir.

Bit Aralığı	47	46	[45:40]	[39:20]	[19:16]	[15:8]	[7:1]	0
Bit Genişliği	1	1	6	20	4	8	7	1
Değer	'0'	'1'	001000	00000h	X	X	X	1
Açıklama	Başla biti	Aktarım biti	Komut indexi (8)	rezerve	Voltaj (VHS)	Kontrol şablonu	CRC7	Dur biti

Voltaj (VHS)	Açıklama
0000	Tanımsız
0001	2.7-3.6V
0010	Düşük voltaj aralığı için rezerve
0100	Rezerve
1000	Rezerve
Diğerleri	Tanımsız

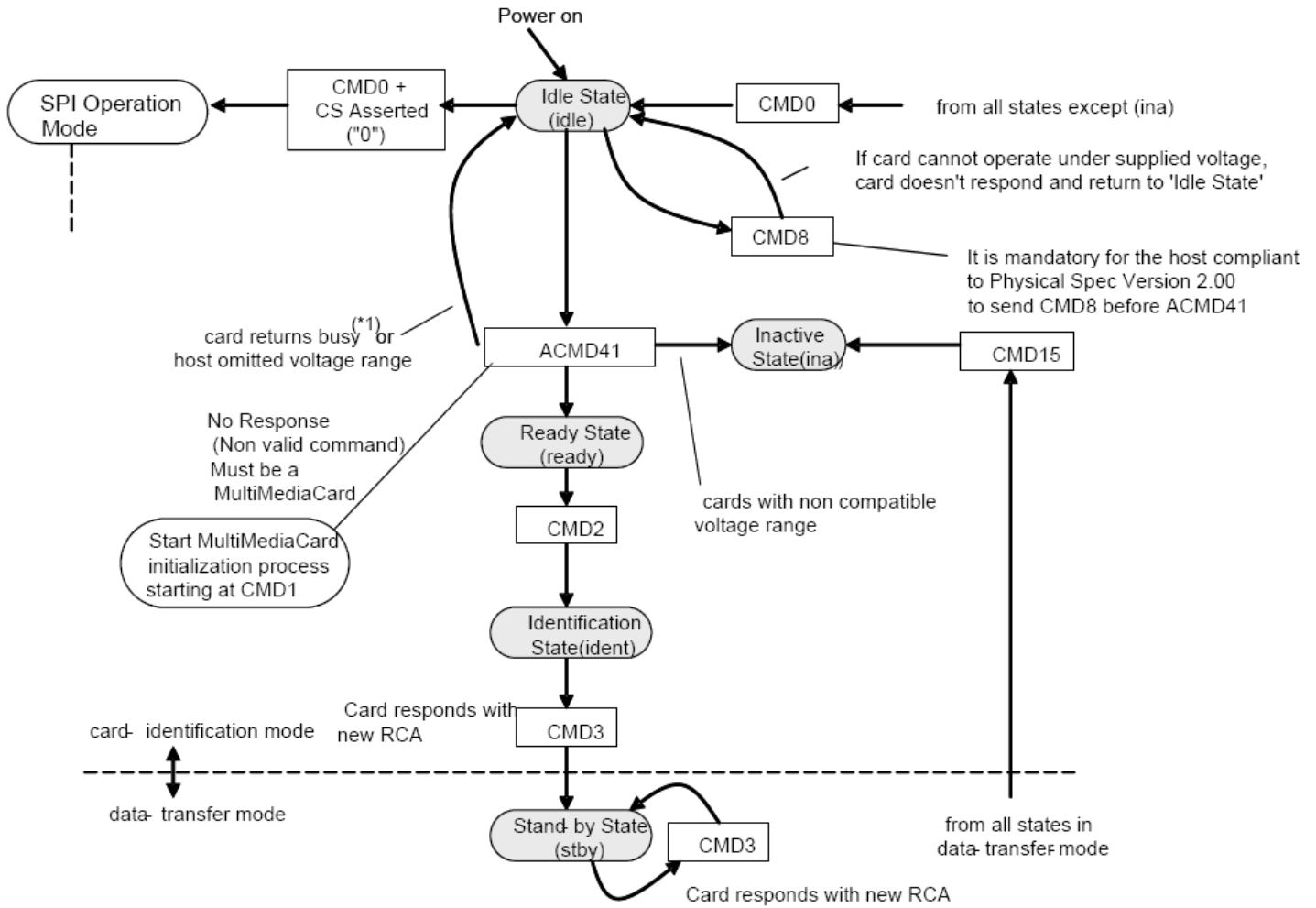
Tablo 2.12 CMD8 (SEND_IF_COND) in içeriği

CMD8 komutunu yanıtlayan kart R7 formatında yanıtlar. Bu format 48 bitliktir ve CMD8 in bit yapısı ile aynıdır.



Düşük voltajda çalışan işlemciler ve SDHC kartlar için ACMD41 komutundan önce CMD8 komutunun gönderilmesi gerekmektedir. Düşük voltajda çalışan işlemciler CMD8 komutunu göndermese çift voltaj aralığına sahip kartlar varsayılan olarak yüksek voltaj aralığında çalışacağından ACMD41 komutunu aldıklarında pasif (inactive) durumuna geçerler.

SD_SEND_OP_COND (ACMD41) komutu V_{DD} çalışma aralığına uyan kartların tanınması, uymayanların ise reddedilmesini sağlamak amacıyla geliştirilmiştir. İşlemci tablo 2.10 da belirtilen çalışma aralığını AMCD41 de parametre olarak karta gönderir. Kart bu aralıkta çalışmayacaksa pasif duruma (inactive) geçer ve kendini sistemden ayırır. Veri yolunda bundan sonra gerçekleşecek olan haberleşmelere katılmaz. Yukarıda anlatılanlar aşağıdaki durum diyagramında gösterilmiştir.



Şekil 2.8 Kimlik doğrulama durum diyagramı

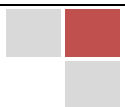
2.4.3 Kart kurulum ve kimlik doğrulama işlemi

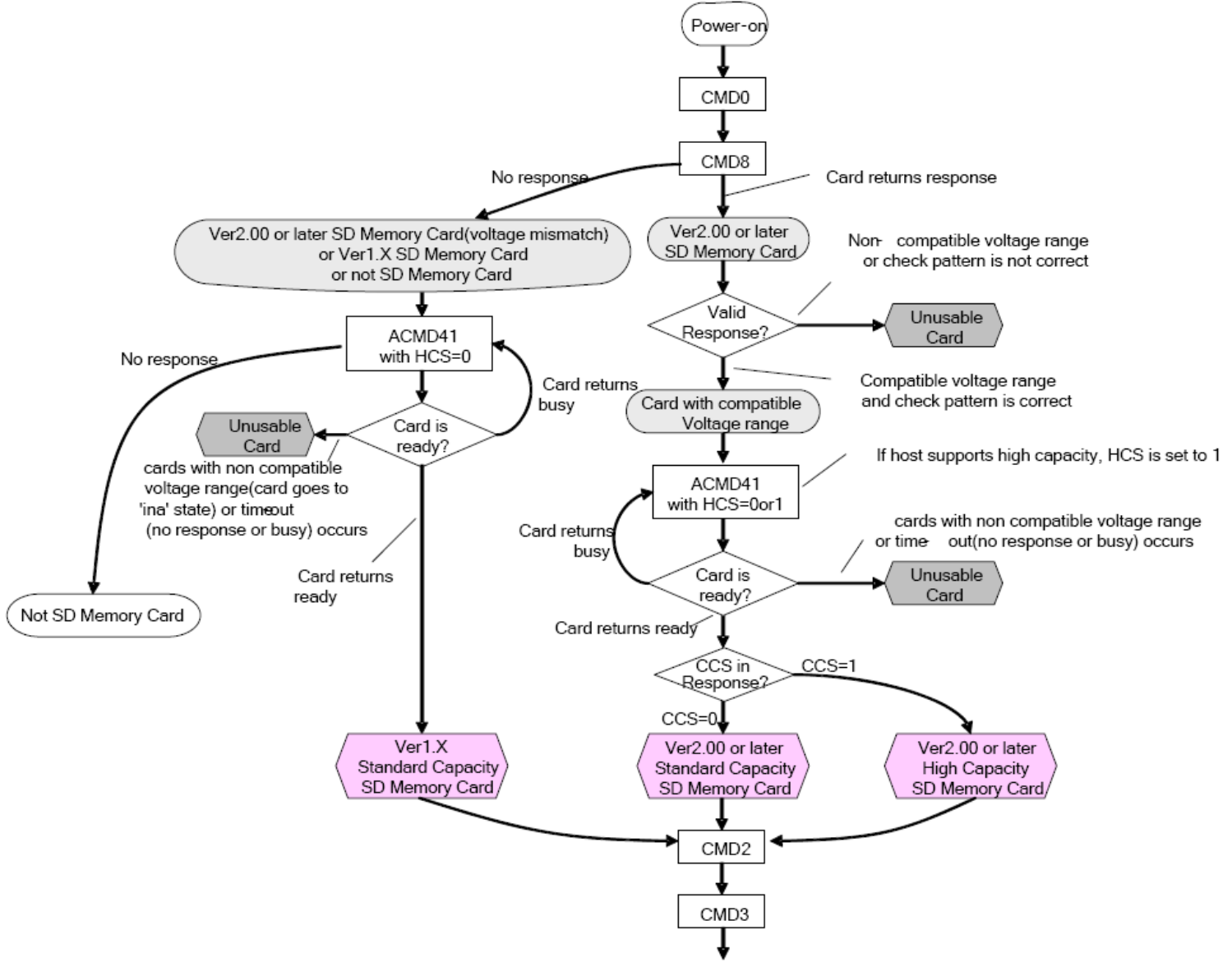
Veri yolu aktifleştikten sonra işlemci kart kurulumu ve kimlik doğrulama işlemlerini başlatır. Kurulum işlemi işlemcinin SD_SEND_OP_COND (ACMD41) ile çalışma koşullarını ve OCR saklayıcısındaki HCS bitini karta göndermesiyle başlar. HCS (Host-İşlemci Capacity Support) biti işlemci SDHC kartları destekliyorsa 1, aksi halde 0 olmalıdır. Kartın CMD8 komutunu cevaplama ACMD41 fonksiyonunu desteklediğini bildirir. ACMD41 komutu gönderilirken HCS biti komut parametresinde gönderilir. Gelen cevapta ise OCR saklayıcısındaki CCR biti okunur. HCS biti CMD8 komutunu cevaplamayan kartlarda görmezden gelinir. Bu durumda işlemci HCS bitini 0 yapmalıdır. Aynı zamanda standart kapasite SD kartlar HCS bitini görmezden gelir. Eğer HCS biti 0 iken SDHC kartına ACMD41

komutu gönderilirse işlemcinin SDHC kartını desteklemeyeceği için OCR deki meşgul (busy)biti 0 a kurulur ve kart hazır durumuna hiç bir zaman geçmez.Bu durumda işlemci kartın kullanışsız olduğuna karar verebilir. Meşgul bitinin 0 olması kartın kurulum (initialization) işlemlerinin devam ettiğini, 1 olması kurulum işlemlerinin bittiğini bildirir. İşlemci meşgul biti 1 olana kadar ACMD41 komutu gönderebilir. Bu esnada ACMD41 komutu ve CMD0 komutu harici bir komut gönderilmemelidir.

Kart CMD8 komutunu cevapladıysa ACMD41 cevabındaki CCS biti bize kartın kapasitesi hakkında bilgi verecektir. CCS biti 1 ise kart 4Gb-32Gb aralığında SDHC karttır. CCS biti 0 ise kart 2Gb a kadar standart bir SD karttır.

İşlemci veriyolundaki tüm kartlar için aynı işlemleri tekrar etmelidir. Şartlara uyan kartlar kimlik doğrulama modu için hazır hale gelirler, uyumsuz kartlar pasif (inactive) durumuna itilirler.Kimlik doğrulama için işlemci kartlara ALL_SEND_CID (CMD2) komutunu gönderir ve kart ID numaralarını bekler. Bu işlemler veriyolundaki CMD hattı üzerinde gerçekleşir. CMD2 komutunu alan kart CMD hattı üzerinden R2 cevabı ile CID saklayıcısını gönderir ve kimlik doğrulama moduna geçer. İşlemci daha sonra kimlik doğrulama moduna geçmiş karta CMD3 (SEND_RELATIVE_ADDR) komutunu gönderir ve karttan RCA adresini göndermesini ister. Bu komutu alan kart bir RCA adresi üretir ve işlemciye gönderir. Bu adres 16 bitlik bir adrestir ve kısa oluşu nedeniyle CID yerine veri haberleşmelerinde RCA kullanılmaktadır.İşlemci CMD3 komutunu her gönderdiğinde kart yeni bir RCA adresi bildirir.Son üretilen RCA kartın güncel adresidir. CMD3 komutunu alan kart R6 cevabı ile yanıt verir. Anlatılanlar aşağıdaki akış diyagramı ile gösterilmiştir.



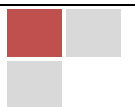


Şekil 2.9 Kimlik doğrulama evresi akış diyagramı

2.5 Veri Transfer Modu (Evresi)

CMD3 komutundan sonra kartlar kimlik doğrulama durumunu geçmiş ve beklemede durumundadırlar. Bu durumda veri haberleşmesine başlamadan önce işlemci kartı 25Mhz lik çalışma frekansına kurabilir. Bu değer SD kartlar için max değerdir. Daha sonra işlemci karta SEND_CSD (CMD9) komutunu göndererek kartın CSD saklayıcısını ister. Bu saklayıcıdan kartın blok uzuluğu hafıza kapasitesi gibi bilgileri alır. CMD7 (SELECT_CARD) komutu RCA adresi verilen bir kartı transfer durumuna sokar. Aynı anda sadece bir kart transfer durumunda olabilir. RCA adresi uyuşmayan kartlar CMD7 komutunu aldığı anda bekleme durumuna geçerler. Eğer işlemci tüm kartları bekleme durumuna sokmak istiyorsa RCA adresini 0X0000 geçer. Bu işlem sistemde kurulu ve beklemekte olan kartları etkilemeden yeni takılan kartların kurulması için kullanılmaktadır. Daha önceden kurulmuş ve RCA adresleri olan kartlar ACMD41 ve CMD2 komutlarını görmezden gelecektir.

Bu basamaklardan sonra gerçekleşecek olan veri transferi sadece adreslenmiş ve transfer durumundaki kart ile işlemci arasında gerçekleşecektir. Tüm haberleşmeler uçtan uca CMD hattı üzerinden yapılır ve aynı şekilde gelen cevap ile onaylanır. Komut ve cevaplar CMD hattı üzerinden



- Kart, CSD saklayıcısını yazma, yazma koruması ve silme komutları için tamponlama mekanizması sağlamaz.Yani bu komutlardan biri işlenirken diğer veri transfer komutları kabul edilmez. DAT0 hattı programlama durumunda kart meşgul olduğu sürece düşük seviyede kalacaktır. Eğer sistemde birden çok kart kullanılacaksa CMD ve DAT0 hatlarını ayrı bağlamak kart meşgul iken diğer kartlar ile haberleşmeye olanak tanır.
- Kart programlama durumundayken parametre kurma komutları kabul edilmez. Bu komutlar: blok uzunluğunu kur (CMD16), blok başlangıcını sil (CMD32) ve blok sonunu sil (CMD33) dir.
- Kart programlama durumundayken okuma komutlarına izin verilmez.
- Başka bir kartı CMD7 ile beklemeden transfer moduna sokmak silme ve programlama işlemlerini durdurmaz.Bu komuttan sonra kart de-select moduna girer. Bu moddayken CMD7 komutu kartı tekrar programlama durumuna sokar.
- Kartı CMD0 ve CMD15 ile resetlemek bekleyen veya süren programlama işlemlerini sonlandırır ve kartın boшта moduna geçmesine neden olur. Veri kaybını önlemek programcının sorumluluğundadır.
- CMD34-37, CMD50 CMD57 komutları gelecekteki herhangi bir genişletme için rezervedir.

2.6 Veri yolu genişliği seçimi

Karta ilk enerji verildiğinde yada CMD0 ile boшта durumuna geçildiğinde kartın yol genişliği 1 bittir. 4 bitlik veri yolu seçimi ACMD6 komutu ile yapılmaktadır. Bu komut için kart kilitli olmamalı ve transfer durumunda olmalıdır.

2.7 Veri Okuma

Normalde veriyolunda bir aktivite yokken DAT hatları pull-up lardan dolayı yüksek seviyededirler.İletişim başlayacağı zaman start bitleri (1 veya 4 bit düşük seviyede) ile birlikte düşük seviyeye çekilirler. Bu sayede veri iletiminin başlayacağı anlaşılır. Başla işaretinden sonra iletilecek veri akımı şeklinde bloklar halinde iletmeye başlar. Dur bitleri (1 veya 4 bit yüksek seviyede) ile veri iletimi durdurulur. İletişim işlemcinin clock darbeleri ile gerçekleştirilmektedir. Veri iletimi sırasında power off kesmesi gelirse, SD kart işlenmekte olan tüm verileri yazma ve silme durumları hariç tüm durumlarda korumayı garanti eder. Yazma veya silme işlemleri sırasında kart çıkarılır yada gücü kesilirse işlenmekte olan veri kaybolur. Veri okuma komutu BLOCK_LEN_ERROR veya ADDRESS_ERROR hataları oluşmuşsa reddedilir ve transfer işlemine başlanmaz.

2.7.1 Blok Okuma

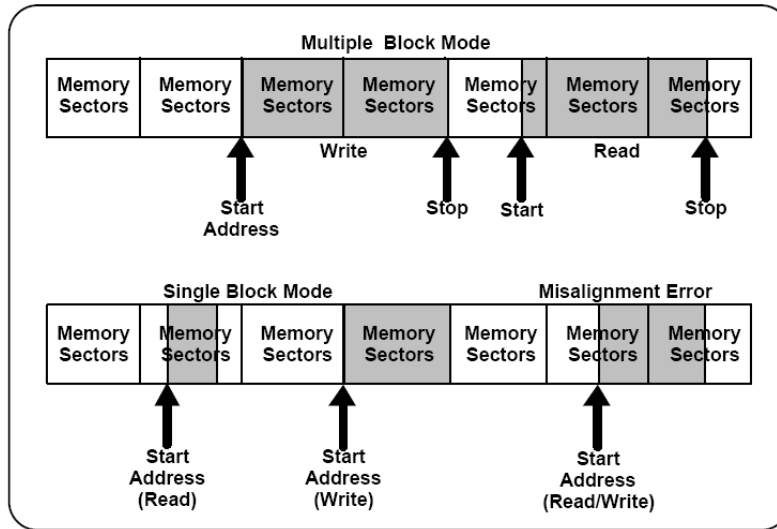
Blok okuma işlemi blok tabanlı bir transferdir ve veri transferinin en küçük elemanı olan bloklar halinde yapılır. Blok uzunluğu maximum 512byte olabilmektedir. Daha kısa bloklarda bir blok uzunluğunun haricinde başlangıç ve bitiş adresi vardır. Bu adresler 512 bytelik blokların sınırları içerisinde olmalıdır. READ_BL_LEN alanı ne olursa olsun blok uzunluğu CMD16 komutu ile max. 512



byte uzunluğa kadar kurulmalıdır. Veri bütünlüğünü korumak için her bloğun sonuna CRC bitleri eklenmektedir. CMD17 komutu (READ_SINGLE_BLOCK) blok okuma işlemini başlatır ve bir blok okunur. Okuma işlemi bittikten sonra kart transfer (tran state) durumuna geri döner. CMD18 ile (READ_MULTIPLE_BLOCK) karttan birkaç ardışıl blok okunabilir. Transfer işlemi CMD12 komutu (STOP_TRANSMISSION) ile durdurulabilir. Transfer işlemi CMD12 komutunun dur bitinin okunması ile durdurulur. Bu arada küçük bir gecikme olmaktadır. Eğer işlemci kısmi bloklar halinde veri okuyacaksa blok hizalı okumalıdır. Okunan aralık blok dışına taşmışta ve kart kaymaya (misalign) izin vermiyorsa ilk kaymış blok tespit edildiğinde kart status saklayıcısındaki ADDRESS_ERROR bitini set eder ve Sending Data State durumunda CMD12 komutunu bekler.

CSD DEĞERİ			Şuanki blok uzunluğu	Oku komutu başlangıç adresi
Max blok uzunluğu READ_BL_LEN	Kayma	Kısmi		
512 byte	0 (pasif)	1 (etkin)	1-512 byte	Herhangi bir adres
1Kb	0 (pasif)	1 (etkin)	1-512 byte	Herhangi bir adres
2Kb	0 (pasif)	1 (etkin)	1-512 byte	Herhangi bir adres

Tablo 2.13 Okuma blok uzunluğu tablosu



Şekil 2.11 Tek ve çoklu blok işlemleri, doğru yazım ve kayma hatası gösterimi

2.8 Veri Yazma

Veri yazma işlemlerinde de veri transfer formatı veri okumadaki gibidir. Veriler bloklar halinde iletilirler ve her bloğun sonuna 1 veya 4 bitlik CRC kodları eklenir. Kart veriyi yazmadan önce CRC bitlerini kontrol eder. Veri yazma komutu BLOCK_LEN_ERROR veya ADDRESS_ERROR hataları oluşmuşsa reddedilir ve transfer işlemine başlanmaz.

2.8.1 Blok Yazma

Veri yazma işlemleri CMD24 - 27, 42, 56(w) komutları ile yapılabilir. Okuma işlemlerinde olduğu gibi yazılacak kısmi blokların adresi CMD16 komutu ile bildirilir. Tablo 2.14 de gösterilmiştir.

CSD DEĞERİ			Şuanki blok uzunluğu	Yaz komutu başlangıç adresi
Max blok uzunluğu WRITE_BL_LEN	Kayma	Kısmi		
512 byte	0 (pasif)	0 (pasif)	1-512 byte	n * 512byte
1Kb	0 (pasif)	0 (pasif)	1-512 byte	n * 512byte
2Kb	0 (pasif)	0 (pasif)	1-512 byte	n * 512byte

Tablo 2.14 Yazma blok uzunluğu tablosu

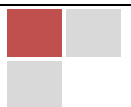
Kısmi blok yazım izni varsa (WRITE_BL_PARTIAL==1) 1byte çözünürlüğünde yazım işlemleri yapılabilir. CRC hatası oluştuğunda kart DAT hatları üzerinden bize bu durumu bildirmektedir, okunan veri çöpe atılır ve çoklu blok yazımı ile veri yazıyorsak takip eden bloklar dikkate alınmaz.Çoklu blok yazımı büyük verilerde yazma işlemlerini hızlandırmak için kullanılmaktadır. Veri yazımı yapılacağı zaman yazılacak verinin başlangıç adresi 512 nin katları şeklinde olmalıdır. Tablo 2.14 ten bu durumu gözleyebilirsiniz.

Eğer kart yazımı kısmi bloklar halinde yapıyorsa ve kayan bloklara izin yoksa (CSD saklayıcısı - WRITE_BLK_MISALIGN==0) kart ilk kayan bloğu tespit ettiğinde ADDRESS_ERROR bitini set eder, okunan verileri ve takip eden verileri göz ardı eder ve programlama durumu yerine receive data durumunda CMD12 (STOP_TRANSMISSION) komutunu bekler. Eğer kayma ilk gönderilen bloktaysa kart transfer durumunda bekleme devam eder, yaz komutuna cevap olarak ADDRESS_ERROR döner. Yazma işlemi aynı zamanda korumalı bir bölgeye yazılmaya çalışıldığında durdurulur. Bu durumda kart WP_VIOLATION bitini set eder.

Bazı kartlar uzun ve tahmin edilemeyen yazım sürelerine sahiptirler. Veri alınıp CRC doğrulaması yapıldıktan sonra kart verileri hafızaya programlamaya başlar. Bu esnada DAT0 hattı düşük seviyeye çekilir. Bunun anlamı yazım tamponu dolu ve kart verileri hafızaya yazmakla meşguldür, yeni veriler kabul edilmezdir. İşlemci herhangi bir zamanda kartın durumunu CMD13 (SEND_STATUS) komutu ile öğrenebilir. READY_FOR_DATA biti yazım işleminin bitip bitmediğini, kartın yeni veri alıp alamayacağını bildirir. İşlemci bir karta yazım işlemi sürerken CMD7 komutu ile kartın seçimini kaldırıp başka bir kartı seçebilir. Bu durumda veri yazmakla meşgul olan kart DAT0 hattını bırakır ve kendi işlemine devam eder. Bu sayede işlemci diğer kartları seçip onlara veri yazabilir. Seçimi kaldırılmış veri yazmakla meşgul olan bir kart tekrar seçildiğinde veri yazma işlemi bitmemiş ise tekrardan DAT0 hattını düşük seviyeye çekecektir. İşlemcinin meşgul kartların seçimini kaldırarak CMD ve DAT0-3 hatlarını kullanarak birden fazla kartı bu şekilde yönetmesine interleaving denilmektedir.

2.8.2 Çoklu Blok Yazımı Öncesi Ön Silme İşlemi

Çoklu blok halinde veri yazmadan önce ACMD23 komutu ile yazılacak blokların önceden silinmesi çoklu blok yazımı işlemine hız kazandırmaktadır. İşlemci bu komutu göndererek bir sonraki yazma işleminde ne kadar blok yazılacağını karta bildirir. İşlemci tüm veri karta gönderilmeden önce CMD12 komutu ile transferi durdurursa kalan bloklardaki veri tanımsız olur (Silinmiş yada eski veriyi taşıyor olabilir). Eğer işlemci ACMD23 komutunda bildirdiğinden fazla blok gönderirse yeni alınan bloklar için kart blok blok silme işlemi uygular. Bu sayı çoklu blok yazımından sonra varsayılan değer olan 1 e kurulmalıdır. Ön silme işleminin zorunluluğu yoktur fakat çoklu blok yazımı komutu (CMD25) öncesinde ne kadar blok yazılacağını bildirilmesi ile yazılacak bloklar ön silme işlemine tabi tutulur. Buda bize hız kazandırır. Eğer bildirim yapılacaksa ACMD23 komutu yaz (CMD25) komutundan hemen önce gönderilmelidir. Aksi halde başka bir komut gönderildiğinde ön silme sayısı resetlenir.



2.8.3 Yazılan Blok Sayısı

Çoklu blok yazımı esnasında bir hata oluşmuş ise işlemcinin bundan haberi olmayabilir. Veri yazımının sonrasında karta ACMD22 komutu gönderirsek kart cevap olarak doğru olarak yazılan blok sayısını verecektir. Bu sayı ilede kontrollerimizi yapıp hangi blokta hata oluştuğunu tespit edip kalanları tekrar gönderebiliriz.

2.9 Silme

Veri akışını arttırmak için kart yazacağı blokları silmelidir. Bunun için bir dizi halinde 3 komut kullanılmaktadır. Bu komutlar silinecek başlangıç adresini bildiren CMD32 (ERASE_WR_BLK_START), son adresi bildiren CMD33 (ERASE_WR_BLK_END) ve son olarak sil komutu olan CMD38 (ERASE) dir. Sil veya adres kurma komutları sıraya uymadan alınırsa eğer kart status saklayıcısındaki ERASE_SEQ_ERROR bitini set eder ve tüm diziyi resetler. Ayrıca dizi dışında bir komut alındığında kart status saklayıcısındaki (ERASE_RESET) bitini set eder ve son alınan komutu işletir.

Eğer silme aralığı yazma korumalı sektörlerle denk geliyorsa bu sektörler dokunulmadan geçilir ve koruması olmayan sektörler silinir. Bu durumda status saklayıcısındaki (WP_ERASE_SKIP) biti set edilir. Adres kurma komutlarındaki adres alanındaki bilgi yazılacak blokların byte cinsinden değeridir. Kart CSD saklayıcısındaki WRITE_BL_LEN altındaki değerleri gözardı etmektedir. Yazma işlemlerinde anlatıldığı gibi silme işlemi esnasında da kart DAT0 hattını süşük seviyeye çekip meşgul olduğunu bildirir. Bu esnada işlemci CMD7 komutu ile başka kart seçebilir.

2.10 Kart Yazma Koruması

SD kartlarda 3 adet yazma koruma mekanizması vardır. Bunlar,

- Kartın yanındaki mekanik switch (İşlemcinin sorumluluğunda)
- Kart dahili yazma koruması (Kartın sorumluluğunda)
- Şifre korumalı kart kilitleme işlemi

Kartın yanındaki mekanik switch ile kullanıcı kartı yazmaya ve silmeye karşı kilitlemek istediğini bildirebilir. Fakat kart bu switchin konumunu bilemez ve kart tarafından takip edilmez. Switchin konumu işlemci tarafından belirlenmelidir.

2.11 Kartı kilitleme işlemleri

Bu kısım kartı kullanıcı tanımlı bir şifre ile kilitleme ve yine aynı şifreye açma imkanının olduğunu belirtmek için yazılmıştır ama detayına burada girilmeyecektir. Detaylı bilgi için datasheetin 4.3.7 bölümüne bakabilirsiniz.

Şifre ile koruma özelliği sayesinde işlemci kartı yazmaya ve okumaya karşı kilitleyebilir. Şifre kullanıcı tanımlıdır. Bir şifre ile kartı şifreledikten sonra aynı şifre ile açabiliriz. Kilitli kartlar temel komut seti olan class0 komutlarına, ACMD41 ve CMD16 komutlarını cevaplayabilirler. Bu sayede kartı resetleyebiliriz, kurulumunu gerçekleştirebiliriz ve durumunu öğrenebiliriz. Fakat veri yazamaz ve okuyamayız. Şifre saklayıcıları ucucu değildir, yani bir kart kilitlenmiş ise power on esnasında kilit otomatik olarak devreye girer. Kartı sistemden çıkarıp takmak bir fayda sağlamaz. Kilitleme ve kilit açma işlemleri için datasheetin 4.3.7 bölümünü inceleyebilirsiniz.

2.12 Komutlar ve Cevaplar

2.12.1 Komut Formatı

Komutlar 48 bit uzunluğundadır ve iletimi için $1.92\mu\text{s}$ (@ 25 MHz) veya $0.96\mu\text{s}$ (@ 50 MHz) gereklidir. Bir komutun yapısı aşağıdaki gibidir.

Bit Aralığı	47	46	[45:40]	[39:8]	[7:1]	0
Bit Genişliği	1	1	6	20	7	1
Değer	'0'	'1'	X	X	X	'1'
Açıklama	Başla biti	Aktarım biti	Komut indexi	parametre	CRC7	Dur biti

Tablo 2.15 Standart komut yapısı

Komutlar her zaman başla biti olan 0 ile başlar ve ardından verinin yönünü bildiren aktarım biti gelir. Biz komut gönderdiğimiz için 1 yapıyoruz, yani işlemciden karta. Ardından 6 bitlik komut indexi gelir. 6 bit ile 2^6 dan 0-63 aralığında 64 adet komut bildirilebilir. Komut indexinden sonra komutun parametre alanı gelir ve bu alan 32 bit genişliğindedir. Bu alanda komut ile birlikte komuta özel parametre karta taşınır. Ardından 7 bitlik CRC kontrolü ile dur biti olan '1' gelir.

Komutların detaylı tanımı için datasheetteki bölüm 4.7.4 altında verilen tabloya bakabilirsiniz.

2.13 Cevap Formatları

Cevaplarda komutlar gibi CMD hattı üzerinden iletilmektedirler. Kod uzunluğu cevap tipine göre değişmektedir. İlk biti her zaman başla biti olan 0, son biti ise dur biti olan 1 dir. 46. biti ise verinin karttan işlemciye taşındığını bildiren 0 dır. R3 tipi hariç tüm cevaplar CRC bitleri ile korunmaktadır. Standart SD kartlar için 5 adet cevap tipi vardır. Bunlar R1, R1b, R2, R3, R6, R7 dir.

R1:

Kod uzunluğu 48 bittir. Formatı komut formatına benzemektedir. Komut indexi alanında hangi komuta cevap olarak geldiği bildirilir. Kart durumu alanında kartın 32 bitlik durum bilgisi okunmaktadır.

Bit Aralığı	47	46	[45:40]	[39:8]	[7:1]	0
Bit Genişliği	1	1	6	20	7	1
Değer	'0'	'0'	X	X	X	'1'
Açıklama	Başla biti	Aktarım biti	Komut indexi	Kart durumu	CRC7	Dur biti

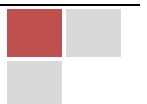
Tablo 2.16 R1 cevap formatı

R1b:

Formatı R1 deki gibidir. Ekstra olarak kartın meşgul olduğunu bildiren b biti cevaptan sonra eklenir. İşlemci kartın meşgul olup olmadığını anlamak için b bitini değerlendirmelidir.

R2 (CID, CSD Saklayıcıları):

Kod uzunluğu 136 bittir. Bu cevap ile kartın CID ve CSD saklayıcıları işlemciye taşınır. Kart CMD2 ve CMD10 komutlarına cevap olarak CID saklayıcısını, CMD9'a cevap olarak CSD saklayıcısını işlemciye R2 ile gönderir. Saklayıcılar [127:1] aralığında taşınır.



Bit Aralığı	135	134	[133:128]	[127:1]	0
Bit Genişliği	1	1	6	127	1
Değer	'0'	'0'	'111111'	X	'1'
Açıklama	Başla biti	Aktarım biti	Rezerve	CID veya CSD (CRC dahildir)	Dur biti

Tablo 2.17 R2 cevap formatı

R3 (OCR saklayıcısı):

Kod uzunluğu 48 bittir. OCR saklayıcısının içeriği AMCD41 komutuna cevap olarak gönderilir.

Bit Aralığı	47	46	[45:40]	[39:8]	[7:1]	0
Bit Genişliği	1	1	6	20	7	1
Değer	'0'	'0'	'111111'	X	'111111'	'1'
Açıklama	Başla biti	Aktarım biti	rezerve	OCR Saklayıcısı	rezerve	Dur biti

Tablo 2.18 R3 cevap formatı

R6 (RCA):

Kod uzunluğu 48 bittir. CMD3 komutuna cevap olarak gönderilmektedir. Parametre alanının üst 16 biti en son yayınlanan RCA adresini alt 16 biti ise kartın durumu hakkında bilgi veren bitlerdir. Bu bitler (23,22,19,12:0) için datasheet tablo 4.35 e bakabilirsiniz.

Bit Aralığı	47	46	[45:40]	[39:8]	[7:1]	0	
Bit Genişliği	1	1	6	16	16	7	
Değer	'0'	'0'	'000011'	X	X	X	
Açıklama	Başla biti	Aktarım biti	Komut indexi (3)	[31:16] RCA	[15:0] durum bitleri	CRC7	Dur biti

Tablo 2.19 R6 cevap formatı

R7:

Kod uzunluğu 48 bittir. CMD8 komutuna yanıt olarak kartın çalışma şartlarını taşır. [19:16] bitleri kartın desteklediği voltaj aralığını bildirir. Çalışma aralığını kabul eden kart R7 formatında voltaj aralığını ve işlemcinin gönderdiği kontrol şablonunu aynen geri döner. CMD8 komutunun yapısını tablo 2.12 de görebilirsiniz.

Bit Aralığı	47	46	[45:40]	[39:20]	[19:16]	[15:8]	[7:1]	0
Bit Genişliği	1	1	6	20	4	8	7	1
Değer	'0'	'0'	001000	00000h	X	X	X	1
Açıklama	Başla biti	Aktarım biti	Komut indexi (8)	rezerve	Voltaj (VHS)	Kontrol şablonu	CRC7	Dur biti

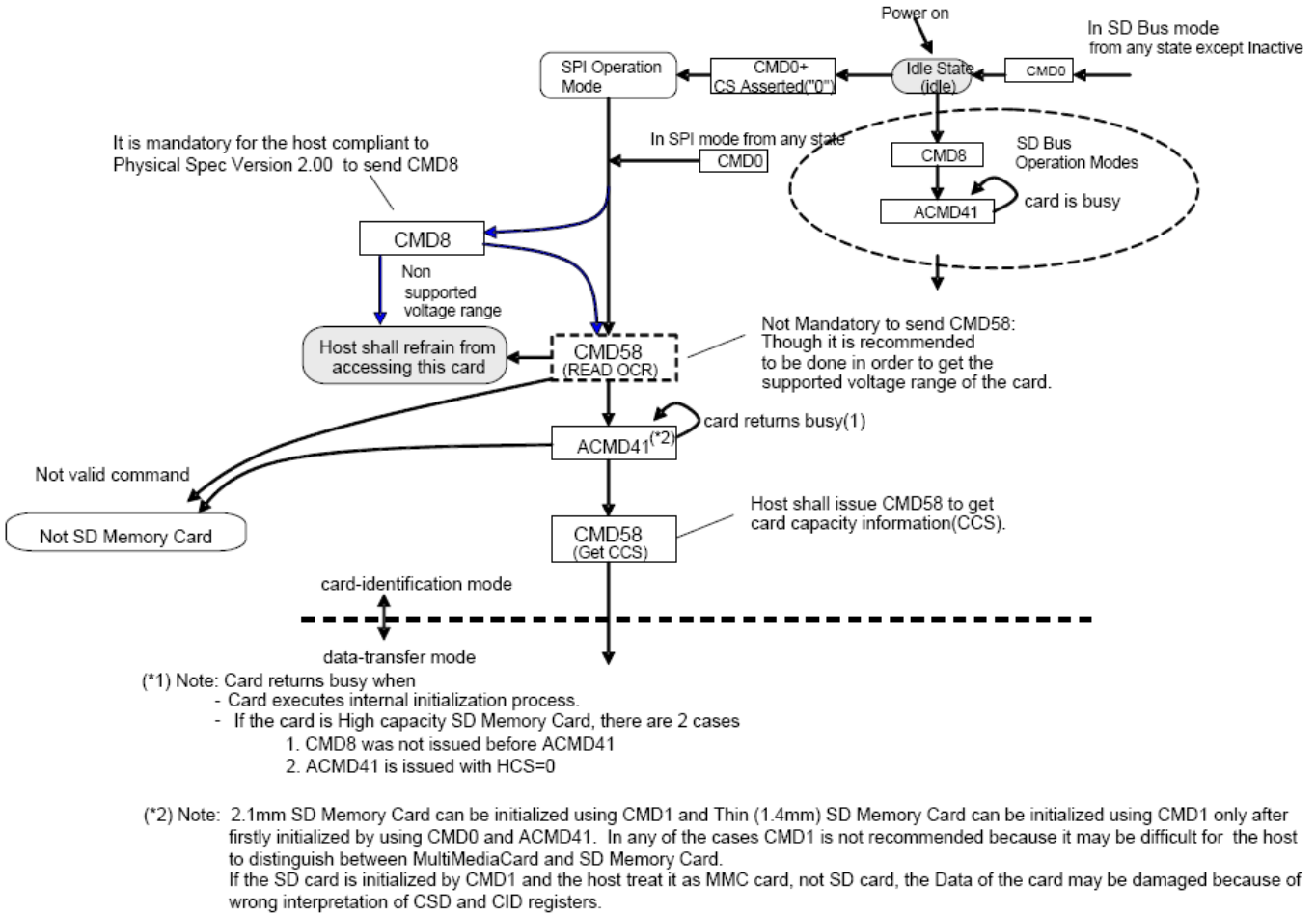
Tablo 2.20 R7 cevap formatı

3. SPI Veri Yolu Kullanımı

SPI, SD kart ile haberleşmede kullanılan ikinci bir moddur. Bu modda haberleşebilmek için power-on ardından CMD0 komutunu gönderirken CS pini düşük seviyede olmalıdır. Bu sayede kart native SD modundan SPI moduna girer ve birdaha çıkmaz, bundan sonraki tüm işlemler SPI üzerinden gerçekleşir. SPI ile haberleşmede SD modunda olduğu gibi komut, cevap ve veri tabanlıdır. Tüm haberleşme 8 bitlik byte lar halinde yapılmaktadır ve CS sinyaline göre hizalanmıştır. CS pini düşük seviyeye çekildiğinde kart master dan gelen saat darbelerini saymaya başlar ve böylece veriler iletilir. Veri uzunlukları 8 saat darbesinin katları şeklinde olmalıdır. Seçili kart bir komut aldığı anda native SD modunun aksine her zaman bir cevap ile karşılık vermektedir.

Okuma işlemi sırasında kart bir hatayla karşılaştığında SD modundaki zaman aşımı yerine hata cevabı ile karşılık vermektedir. Aynı zamanda yazım işlemleri sırasında karta gönderilen her blok sonrasında kart karşılık vermektedir.

Standart kapasiteli SD kartlarda veri blok uzunluğu en fazla "write block" uzunluğu kadar, en küçük bir byte kadar olabilir. Kısmi yazım ve okuma işlemleri CSD saklayıcısındaki seçeneklerle yapılabilir. Yüksek kapasiteli SDHC kartlarda veri bloğu 512 byte a sabitlenmiştir. Blok uzunluğu sadece kart kilitleme işlemleri için CMD42 (LOCK_UNLOCK) komutundan önce CMD16 (SET_BLOCKLEN) komutu ile ayarlanmaktadır. Bu yüzden kısmi yazma ve okumaya izin verilmez. Ayrıca CMD28 (SET_WRITE_PROT), CMD29 (CLR_WRITE_PROT) ve CMD30 (SEND_WRITE_PROT) komutlarında SDHC kartlarda desteklenmemektedir.

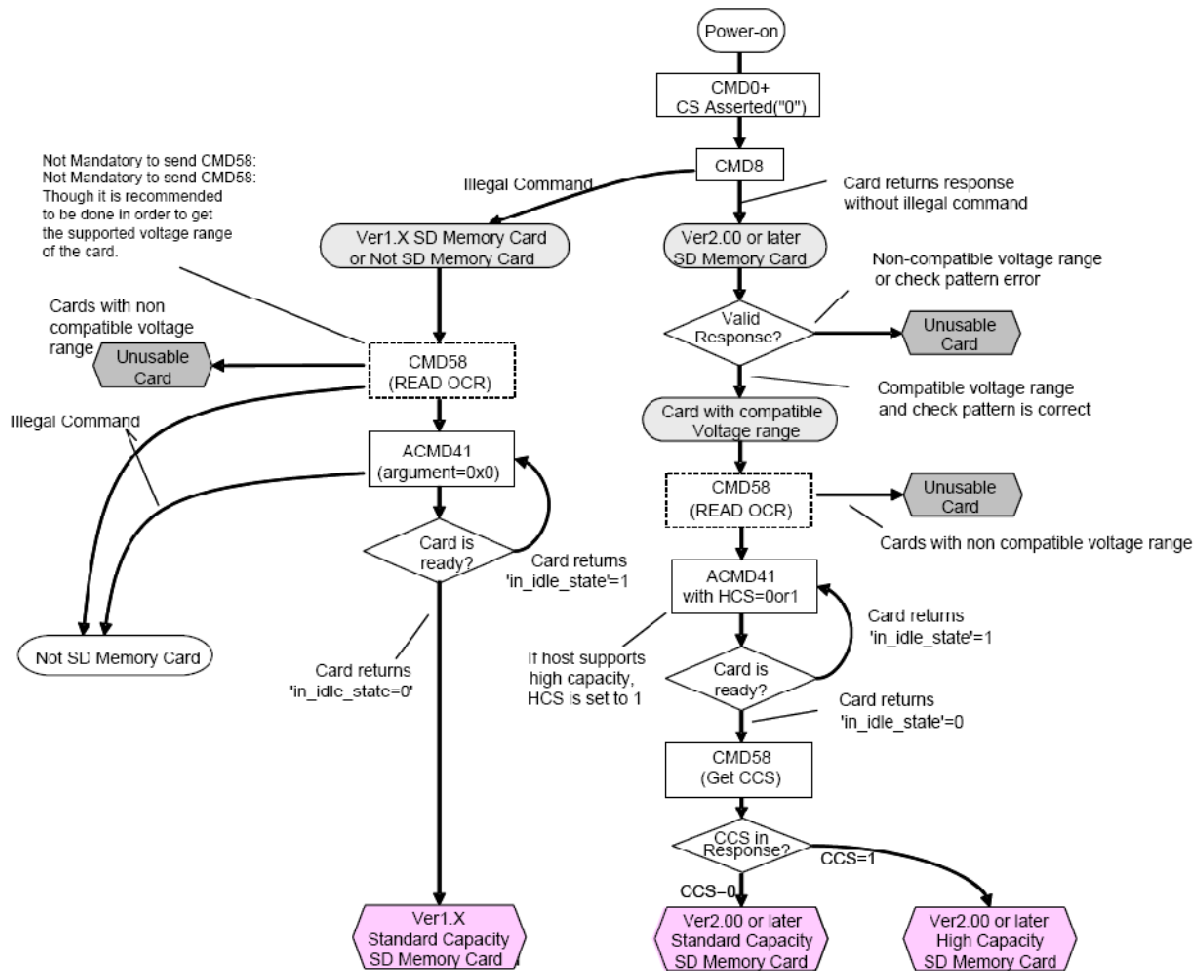


Şekil 3.1 SD kart durum diyagramı (SPI modu)

3.1 Mod Seçimi ve Kurulum

SD kart power-up esnasında native (SD modu) modunda açılmaktadır. Power-on evresinden sonra karta reset komutu olan CMD0 komutunu gönderirken CS pinini düşük seviyede olursa kart ikincil haberleşme modu olan SPI moduna girmektedir ve tüm haberleşme işlemlerini SPI modunda devam ettirmektedir. SD moduna geri dönüş için power reset atmanız gerekmektedir. Kart SPI modunda haberleşmeye devam edileceğini anladığında SPI moduna özel R1 cevabı ile karşılık vermektedir. SPI cevap formatları farklıdır ve ileride anlatılacaktır.

SD modunda olduğu gibi SPI modunda da önce işlemci ve kartın çalışma gerilimlerinin eşleşip eşleşmediğinin kontrolü yapılmalıdır. Bunu SD modunda olduğu gibi CMD8 (SEND_IF_COND) komutu ile yaparız. Kart gönderilen veriyi kontrol eder ve R7 (şekil 3.12) formatında cevap verir. İşlemcide dönen cevabı kontrol etmelidir. Karta gönderilen verinin parametre kısmının [11:8] aralığı VHS bitlerini içermektedir. Bu bitlerden aynı anda sadece bir tanesi 1 olabilir. [7:0] alanı kontrol şablonudur ve işlemci karta ne gönderdiyse gelen cevapta aynısını geri okumalıdır. Uygulama için 10101010 değeri tavsiye edilmektedir. CMD8 komutunun yapısını tablo 2.12 de görebilirsiniz. Karttan dönen cevapta VCA 0 ise kartın istenilen besleme geriliminde çalışmayacağı anlaşılır. Ayrıca gönderilen ve gelen kontrol şablonu tutarsız ise haberleşmede hata oluşmuş demektir. Bu durumda CMD8 tekrar gönderilmelidir. Aynı zamanda CMD8 komutuna yanıt veren kart SD v2.0 veya üzeri bir karttır. Eğer cevap olarak geçersiz komut (Illegal Command) hatasını alıyorsak kart SDv1.x kartıdır ifadesini çıkarabiliriz.



Şekil 3.2 SPI modu kurulum akış diyagramı

Kart kurulum işlemini CMD1 komutu ilede yapabiliriz fakat bu komut için bazı şartlar vardır. Bu şartlar kartın ince (1.4mm) yapıda standart kapasitede SD kart olması ve kurulum işleminin tekrar yapılıyor olmasıdır. Yani kart ilk enerjilenip çalışmaya başladığı durumda (poer-on) CMD1 komutu kullanılmamalıdır.

CMD58 (READ_OCR) komutu ile işlemci kartın işlemci tarafından belirlenen V_{DD} geriliminde çalışıp çalışmayacağı anlaşılabilir. Aynı zamanda kartın SD kartı SDHC kartı olduğu OCR saklayıcısının okunması ile anlaşılabilir. Detaylı bilgi için bölüm 2.2.4 ve tablo 2.10 a göz atabilirsiniz. Kartın voltajı işlemci tarafından kabul edilmezse kart kullanışsız kart olarak işaretlenir ve göz ardı edilir. Şekil 3.2 den görebilirsiniz.

ACMD41 (SD_SEND_OP_COND) komutu kart kurulum (başlatma) işlemlerinin başlatılması için ve işlemlerin tamamlanıp tamamlanmadığını öğrenmek için gönderilmektedir. CMD8 komutunun ACMD41 komutundan önce gönderilmesi şarttır. CMD8 komutunu alan kart CMD58 ve ACMD41 fonksiyonlarını genişletmektedir. CMD58'e gelen cevapta CCS biti sınanarak kartın SDHC mi yoksa normal SD kartı olduğu anlaşılabilir (bölüm 2.2.4). Aynı şekilde işlemci eğer SDHC kartları destekliyorsa ACMD41 komutunun parametre alanındaki HCS bitini set etmelidir. HCS biti CMD8 komutunu cevaplamayan kartlarda görmezden gelinir. Bu durumda işlemci HCS bitini 0 yapmalıdır. Aynı zamanda standart kapasite SD kartlar HCS bitini görmezden gelir. ACMD41 komutuna gelen cevabın R1 alanındaki "in idle state" biti kartın işlemciyi haberdar etmesini sağlar. Bu bit 1 ise kart kurulum (başlangıç) işlemlerini devam ettiriyor demektir. Bu durumda işlemci karta 0 okuyana kadar tekrar tekrar ACMD41 komutu göndererek kartın işlemlerini bitirip bitirmediğini anlayabilir. Kart HCS bitini sadece ilk ACMD41 de kontrol eder. Tekrarlayan ACMD41 komutları gönderilirden CMD0 harici başka bir komut gönderilmemelidir.

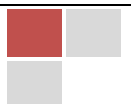
Kurulum işlemleri tamamlandıktan sonra işlemci CCS bitini CMD58 komutuna gelen cevaptan okuyabilir. Bu cevap R3 formatındadır. CCS bitinin değeri ancak kart CMD8 komutunu cevaplamış ve kurulum işlemlerini bitirmiş ise geçerlidir. CCS==1'in anlamı kart SDHC karttır, CCS==0'in anlamı kart SD karttır demektir.

3.2 Veri Yolu Koruması

SD kart modunda tüm komutlar CRC bitleri ile korunmaktadır. SPI modunda CRC kontrolü devre dışıdır. İşlemci komut gönderirken CRC alanını önemsiz (x) olarak geçer ve kart komutu aldığı anda CRC kontrolü yapmaz. Başlangıçta reset komutu (CMD0) haberleşme modunu SD den SPI ya geçirmektedir. Bu komut alındığında kart SD modunda olduğundan CRC bitleri kullanılmalıdır. CMD0 komutunun parametre alanı boş ve kalan bilgileri sabit olduğundan CRC hesaplamasının çalışma esnasında yapılmasına gerek yoktur. CMD0 komutunun yapısı aşağıdaki gibi olmalıdır.

0x40, 0x0, 0x0, 0x0, 0x0, 0x95

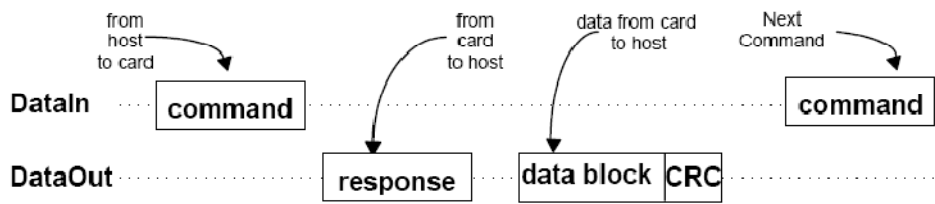
CMD0 komutundan sonra kart SPI moduna geçmiştir ve CRC kontrolü devre dışıdır. Bu esnadan sonra CMD0 dahil tüm SPI komutlarının CRC kontrolü CMD59 (CRC_ON_OFF) komutuna bağlı olarak yapılmaktadır. İşlemci CRC kontrolünü CMD59 komutu ile etkinleştirip devre dışı bırakabilir. CMD59 komutunun parametre alanının 0. biti 1 ise CRC kontrolü etkin, 0 ise pasiftir. Eğer CRC kontrolü etkin ise SPI modunda CRC hesaplama işlemleri kullanıcının sorumluluğundadır. İşlemci CRC kontrolünü etkinleştirecekse bunu ACMD41 komutundan önce yapabilir. CMD8 komutunun CRC



kontrolü devamlı etkindir. İşlemci parametre alanının CRRC bitlerini doğru bir şekilde hesaplamalıdır. Eğer bir CRC hatası oluşmuş ise komut indeksinden bağımsız olarak kart R1 formatında cevap vermekte ve cevabın CRC hatası biti 1 olmaktadır.

3.3 Veri Okuma

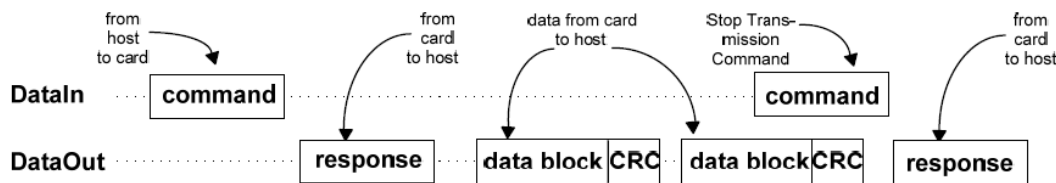
SPI modunda da tek (CMD17) ve çoklu (CMD18) bloklar halinde veri okunabilmektedir. Geçerli bir okuma komutu alındığında kart veriden önce oku komutunu cevaplamakta ve ardından veriyi göndermektedir. Cevap R1 formatındadır. Standart kapasiteli SD kartlarda veri uzunluğu SET_BLOCKLEN (CMD16) ile belirtilen kadardır. SDHC kartlarda block uzunluğu CMD16 dan bağımsız olarak sabit 512 byte tır. Her verinin sonuna 16 bitlik CRC kodları eklenmiştir. CRC standart CCITT polinomu ile hesaplanır. Bu polinom şu şekildedir, $x^{16}+x^{12}+x^5+1$. Aşağıdaki şekilde tek bir blok okuma işlemi görülmektedir.



Şekil 3.3 Tek blok okuma işlemi

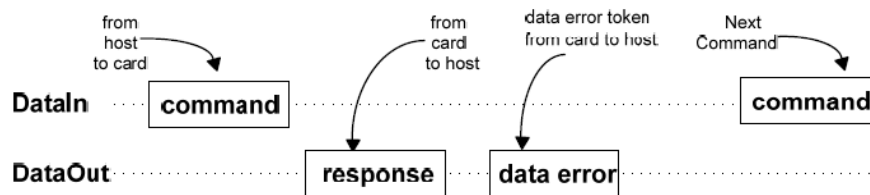
Maximum blok uzunluğu CSD saklayıcısındaki READ_BL_LEN alanından bağımsız olarak 512 byte uzunluğundadır. Standart kapasiteli SD kartlarda kısmi blok okuma (READ_BL_PARTIAL==1) etkinleştirilmişse, blok uzunluğu 1-512 byte arasında herhangi bir sayı olabilir. Başlangıç adresi ise kartın geçerli adres aralığında olmalıdır. Bu durumda okunacak blok aralığı tek bir sektör içerisinde olmalıdır. SDHC kartlarda kısmi blok işlemlerine izin verilmediğinden okunacak blok uzunluğu sabit 512 bytettir ve başlangıç adresi blok sınırlarına hizalı olmalıdır.

Çoklu blok okuma modunda kart blokları ardışıl olarak işlemciye göndermektedir. Her blok, bloktan sonra eklenen kendi CRC16 bilgisi ile korunmaktadır. İşlemci dur komutunu gönderdiğinde (CMD12) kart blok göndermeyi durdurur. Şekil 3.4 te gösterilmiştir.



Şekil 3.4 Çoklu blok okuma işlemi

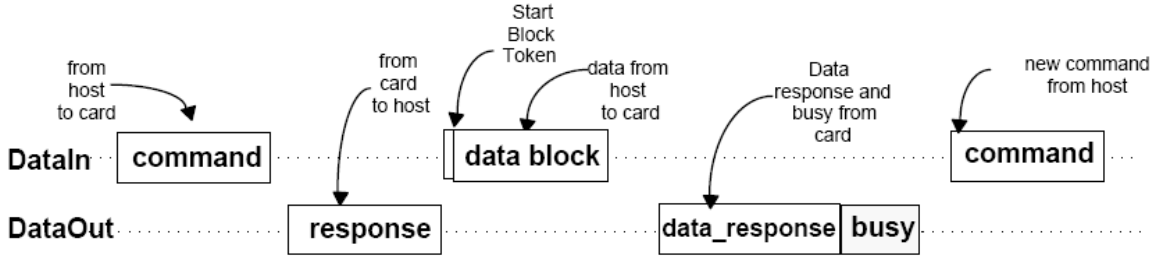
Eğer okuma işlemleri sırasında bir hata ile karşılaşırsa kart veri yerine hata mesajı gönderir ve veri gönderimini durdurur. Şekil 3.5 te hata mesajı ile durdurulmuş okuma işlemi gösterilmiştir.



Şekil 3.5 Hata mesajı ile durdurulmuş okuma işlemi

3.4 Veri Yazma

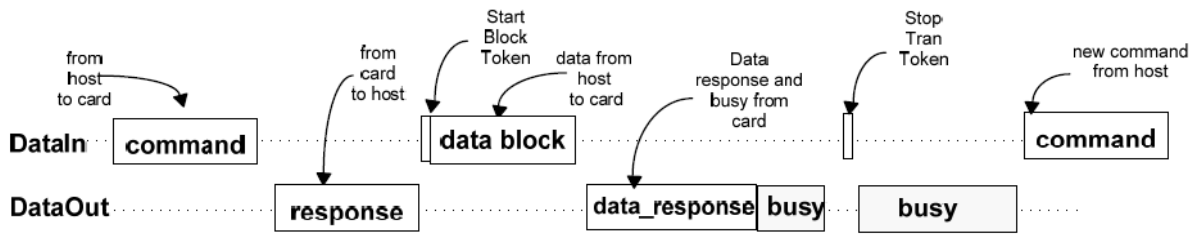
SPI modunda da tek (CMD24) ve çoklu (CMD25) bloklar halinde veri yazılabilmektedir. Geçerli bir yazma komutu alındığında kart veriden önce yaz komutunu cevaplamakta ve ardından veriyi beklemektedir. Cevapladığı format R1 formatındadır. Blok uzunluğu, CRC gibi bilgiler okuma işlemlerinde olduğu gibidir. Şekil 3.6 da yazma işlemi gösterilmiştir.



Şekil 3.6 Tek blok yazma işlemi

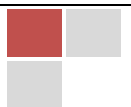
Her veri bloğu blok başlangıcını belirten 1byte'lık başlangıç işareti (Start Block Token) ile başlar. Veri bloğu alındıktan sonra kart data_response ile işlemciye karşılık verir. Kart işlemciden aldığı veriyi hafızaya programlarken meşgul olduğunu bildiren meşgul işaretini işlemciye gönderir (DataOut hattını düşük seviyeye çeker). Kart programlama işlemini bitirdikten sonra işlemci programlama işleminin sonucunu CMD13 (SEND_STATUS) komutu ile öğrenir. Çünkü bazı hatalar (adres taşması, yazma koruması vb.) sadece programlama işlemi esnasında oluşmaktadır. Eğer yazım esnasında bir hata oluşursa (yazma hatası, CRC hatası) kart bunu veri cevabı (data response) ile işlemciye bildirir. Bu cevabın içeriği şekil 3.13 te gösterilmiştir.

Çoklu yazım işleminde veri haberleşmesi dur işareti (stop tran) ile sonlandırılmaktadır. Çoklu yazım işleminde hatalı yazım durumunu öğrenmek için işlemci veri cevabı (data response) yerine ACMD22 (SEND_NUM_WR_BLOCKS) komutunu kullanarak hatasız bir şekilde yazılan blok sayısını öğrenmelidir. Bu sayı ile hata oluşmuşsa kaldığı yerden hatalı verileri göndermeye devam edebilir.



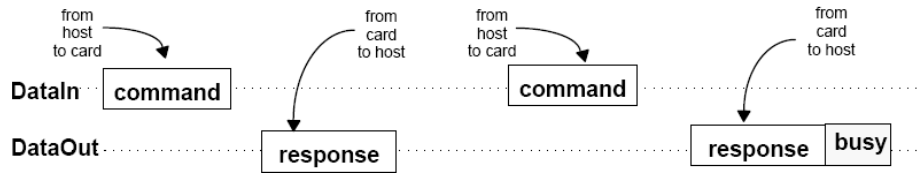
Şekil 3.7 Çoklu blok yazma işlemi

Kart meşgul iken CS sinyalinin kaldırılması kartın yazım işlemlerini etkilemez. Bu durumda kart meşgul olduğunu bildirmek için DataOut pinini düşük seviyede tutuyorken bu pini bırakır. Arka planda yazma işlemleri devam etmektedir. Eğer kart yazım işlemini bitirmeden tekrar CS pini ile seçilirse kart tekrar DataOut pinini düşük seviyeye çekecektir. Eğer yazma işlemi devam ediyorken CMD0 komutu ile kart resetlenirse yazılmakta beklemekte olan veriler kaybolur. Bu durumun oluşmaması kullanıcının (işlemcinin) sorumluluğundadır.



3.5 Silme ve Yazma Koruma Yönetimi

SPI modundaki silme ve yazma koruma işlemleri SD modundaki gibi yapılmaktadır. Kart silme işlemi yaparken yada belirliten sektör listesine ait yazma koruma bitlerini değiştirirken meşgul durumda olacaktır ve DataOut pinini düşük seviyeye çekecektir. Şekil 3.8 de verisiz veri yolu iletişimini görebilirsiniz.



Şekil 3.8 Veri yolu iletişimi

3.6 CID/CSD Saklayıcılarını Okuma

SD modunun aksine (SD modunda bu saklayıcılar gönderilen komutlara dönen cevapların içerisinde) SPI modunda bu saklayıcıların okunması blok okuma ile yapılır. Kart standart veri okuma cevabını (şekil 3.3) gönderdikten sonra CRC16 ile korunmuş 16 byte'lık veri bloğunu gönderir.

3.7 Reset İşlemleri

SD kart önceden belirlenmiş reset dizisi ile resetlenmektedir. Kart power-on evresinden sonra boştaki (idle) durumuna girer ve bu durumda sadece şu komutlar geçerlidir: CMD8 (SEND_IF_COND), ACMD41 (SD_SEND_OP_COND), CMD58 (READ_OCR) and CMD59 (CRC_ON_OFF). Diğer komutlar geçersizdir.

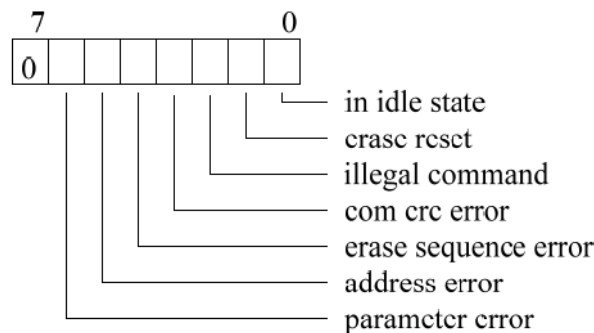
Kalın (2.1mm) SD kartlar için ayrıca CMD1 komutu, ACMD41 komutu ile eşdeğerdir. Fakat genel uyumluluk bakımından CMD1 komutunun kullanılması tavsiye edilmez. İnce (1.4mm) standart kapasiteli SD kartlar için CMD1 komutu geçersiz bir komuttur. Bu komut ancak kartın ilk kurulumu (initialization) ACMD41 komutu ile yapıldıktan sonra, sonraki reset işlemlerinin ardından gerçekleştirilen kurulumlarda geçerlidir.

3.8 Cevaplar

Bu bölümde SPI modu için cevap çerçeveleri anlatılacaktır. SD modunda olduğu gibi en değerli byte önce gönderilmektedir.

3.8.1 R1

Bu cevap her SEND_STATUS komutu hariç her komuttan sonra gönderilmektedir. Bir byte uzunluğundadır ve MSB her zaman 0'dır. Diğer bitleri oluşan hataları temsil etmektedir. Eğer hata oluşmuşsa ilgili bit 1 aksi halde 0 olacaktır.



Şekil 3.9 R1 Cevap formatı

In Idle State: Kart boşta durumundadır ve kurulum (initialization) işlemleri devam etmektedir.

Erase Reset: Silme işlemleri için kullandığımız silme komut dizisi tamamlanmadan arada başka bir komut geldi.

Illegal Command: Geçersiz bir komut kodu tespit edildi.

Com CRC Error: Son komutun CRC kontrolü geçersiz.

Erase Sequence Error: Silme komut dizisi içerisinde bir hata oluştu.

Address Error: Komut içerisinde blok uzunluğu ile uyuşmayan kaymış bir adres kullanıldı.

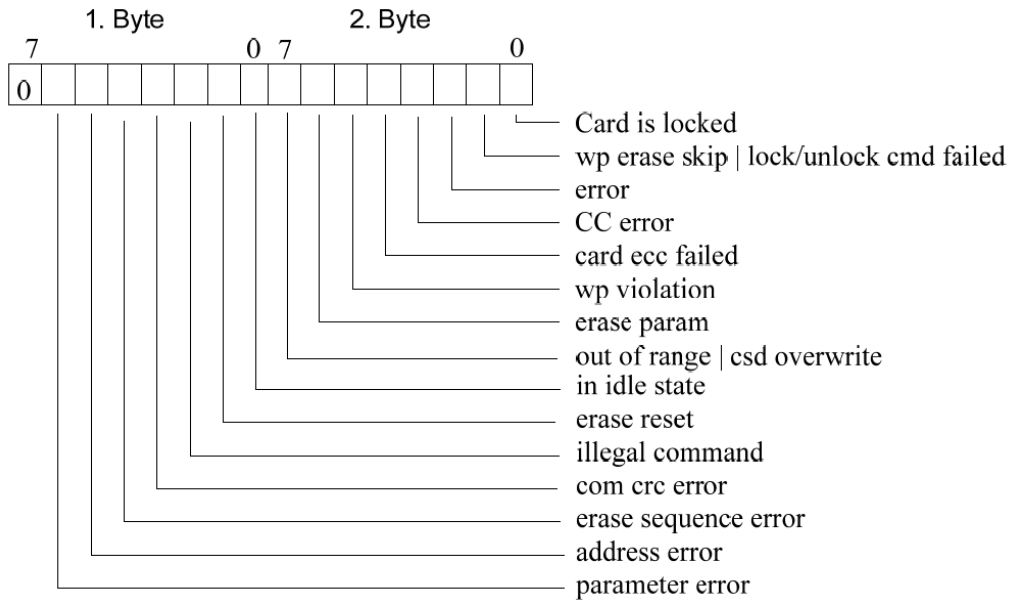
Parameter Error: Komutun parametresi (adres, blok uzunluğu) kart için izin verilen sınırların dışına taşı.

3.8.2 R1b

Bu cevap R1 ile eşdeğerdir sadece R1 den sonra meşgul bilgisi gönderilmektedir. Meşgul bilgisi herhangi bir byte uzunluğunda olabilir. 0 değeri kartın meşgul olduğunu, 1 değeri kartın sıradaki komut için hazır olduğunu bildirir.

3.8.3 R2

Bu cevap iki byte uzunluğundadır ve SEND_STATUS komutuna cevap olarak dönmektedir. Formatı şekil 3.10 da gösterilmiştir.



Şekil 3.10 R2 cevap formatı

Bu formatta cevabın ilk byte'ı R1 ile aynıdır. İkinci byte aşağıda açıklanmıştır.

Erase Param: Silmek için geçersiz bir seçim yapıldı (sektörler, gruplar)

Write Protect Violation: Komut yazma korumalı bir alana yazmaya çalıştı.

Card ECC Failed: Kartın dahili ECCsi kullanıldı fakat veriyi düzeltmedi.

CC Error: Dahili kart kontrolcüsünde hata oluştu.

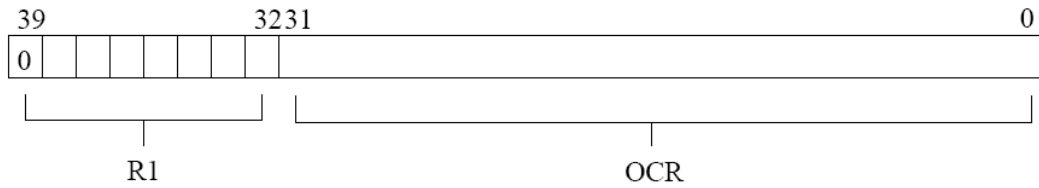
Error: Bilinmeyen veya genel bir hata oluştu.

Write protect erase skip | lock/unlock command failed: Bu bitin iki fonksiyonu vardır. İlki işlemci yazma korumalı bir alanı silmeye çalışıldığında set olur. İkincisi kart kilitleme, kilit açma işlemlerinde komut dizisi yanlış yada şifrede hata oluşmuştur.

Card Locked: Kart kullanıcı tarafından kilitletiğinde set olur.

3.8.4 R3

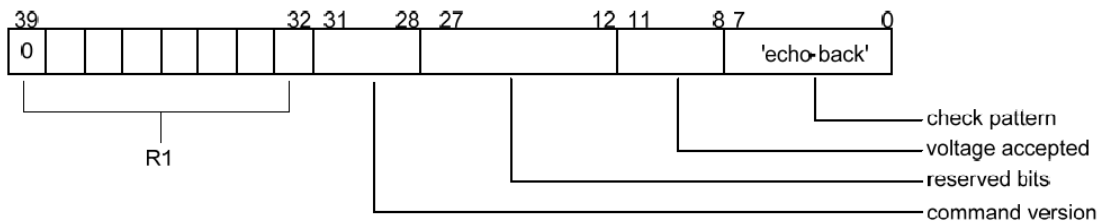
Bu cevap READ_OCR komutu geldiğinde gönderilir. Cevap uzunluğu 5 byte tır. İlk byte R1 ile aynıdır, sonraki 4 byte OCR saklayıcısının değeridir.



Şekil 3.11 R3 Cevap formatı

3.8.5 R7

Bu cevap SEND_IF_COND (CMD8) komutuna cevap olarak dönmektedir. Cevap uzunluğu beş byte tır. İlk byte R1 ile eşdeğerdir. Diğer bilgiler kart çalışma gerilimi, kontrol şablonu, gibi bilgilerdir.

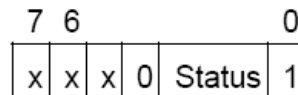


Şekil 3.12 R7 Cevap formatı

3.9 Kontrol İşaretleri

3.9.1 Veri Cevap İşareti (Data Response Token)

Karta yazılacak her veri bu işaret ile onaylanmalıdır. Uzunluğu 1 byte tır ve içeriği şekil 3.13 teki gibidir.



Şekil 3.13 Veri cevap işareti

Status bitlerinin anlamı şöyledir.

010: Veri kabul edildi.

101: Veri CRC hatasından dolayı reddedildi.

110: Veri yazma hatasından dolayı reddedildi.

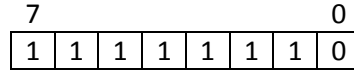
Çoklu blok yazımı esnasında herhangi bir hata oluştuğunda (CRC, yazma) işlemci veri gönderme işlemini CMD12 ile durdurmalıdır. Sorunun ne olduğunu anlamak için CMD13 göndermelidir. Ayrıca ACMD22 ilede doğru bir şekilde yazılan blok sayısını okuyabilir.

3.9.2 Blok Başlangıç ve Transferi Durdur İşaretleri

Veriler okunurken ve yazarken bazı işaretler ile gönderilip alınırlar. Tüm veri byte ları yüksek bittten başlanarak gönderilir. Bu işaretler 4 ten 515 byte uzunluğa kadar olabilirler ve aşağıdaki formattadırlar.

Tek blok okuma, tek blok yazma ve çoklu blok okuma için:

- İlk byte: Başla bloğu

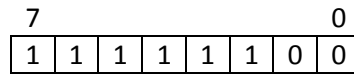


- 2-513 byteleri arası (veri blok uzunluğuna bağlı değişir): Kullanıcı verisi
- Son iki byte :CRC16

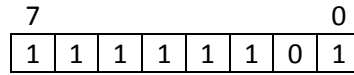
Çoklu blok yazımı için:

Her bloğun ilk byte'ı:

Veri gönderilecekse - Veri bloğu başlangıç işareti



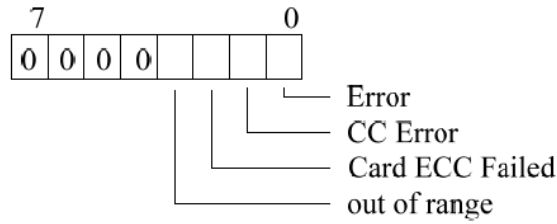
Transfer durdurulacaksa - Transferi durdur işareti



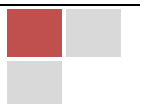
(Bu format sadece çoklu blok yazımında kullanılır. Çoklu blok okumada CMD12 komutu kullanılarak haberleşme durdurulur)

3.9.3 Veri Hata İşareti

Eğer veri okuma işlemi başarısız olursa kart veri yerine bu hata işaretini gönderir. Bu işaret bir byte uzunluğundadır ve aşağıdaki şekilde gibidir. Değersiz 4bit R2 cevabındakilerle aynıdır.



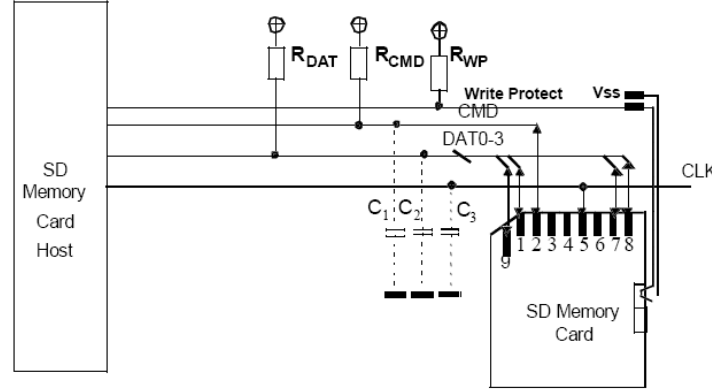
Şekil 3.14 Veri hata işareti



4. Bağlantı

4.1 SD bağlantısı

Kart yuvası boştayken, kartı takıp çıkarırken oluşabilecek akımlara karşı giriş olan pinlerin pull-up yapılması tavsiye edilmektedir. Yazma koruma switch i içinde pull-up direnç kullanılması gerekmektedir. Kullanılacak dirençler 10K-100K arasında olabilir (50K tavsiye edilmektedir). Şekil 4.1 de anlatılan gösterilmiştir. Şekil 4.2 de ise SD modunda kullanılan pinler ve isimleri gösterilmiştir. Bu pinlerden CMD pini seri bir şekilde komut ve cevapları taşımaktadır. DAT0-3 pinleri verileri bir bit veya dört bit şeklinde taşımaktadır. İşlemci ile senkronizasyonu CLK pini sağlamaktadır.



Şekil 4.1 Veri yolu devresi

Pin #	Name	Type ¹	SD Description
1	CD/DAT3 ²	I/O ³	Card Detect/Data Line [Bit 3]
2	CMD	I/O	Command/Response
3	Vss1	S	Supply voltage ground
4	VDD	S	Supply voltage
5	CLK	I	Clock
6	Vss2	S	Supply voltage ground
7	DAT0	I/O	Data Line [Bit 0]
8	DAT1	I/O	Data Line [Bit 1]
9	DAT2	I/O	Data Line [Bit 2]

NOTES: 1) S=power supply; I=input; O=output using push-pull drivers.
 2) The extended DAT lines (DAT1-DAT3) are input on power up. They start to operate as DAT lines after the SET_BUS_WIDTH command. It is the responsibility of the host designer to connect external pullup resistors to all data lines even if only DAT0 is to be used. Otherwise, non-expected high current consumption may occur due to the floating inputs of DAT1 & DAT2 (in case they are not used).
 3) After power up, this line is input with 50Kohm(+/-20Kohm) pull-up (can be used for card detection or SPI mode selection). The pull-up may be disconnected by the user, during regular data transfer, with SET_CLR_CARD_DETECT (ACMD42) command.

Şekil 4.2 SD modundaki pin bağlantıları ve isimleri

4.2 SPI bağlantısı

SPI bağlantısında sekizinci ve dokuzuncu pinler kullanılmamaktadır. Bu pinler pull-up dirençleri ile yükseğe çekilmelidirler. Aksi halde beklenmedik bir yüksek akım tüketimi gerçekleşebilir. Şekil 4.3 te SPI bağlantı pinleri gösterilmiştir. İki numaralı pin ile işlemciden karta (MOSI) komut ve veri aktarımı, yedi numaralı pin ile karttan işlemciye (MISO) veri ve cevap aktarımı gerçekleştirilmektedir. CS pini ile kart ile haberleşmeye başlamadan önce düşük seviyeye çekilmelidir. Böylece haberleşmek istediğimiz kart seçilmiş olur.

Pin #	Name	Type ¹	SPI Description
1	CS	I	Chip Select (Active low)
2	DataIn	I	Host to Card Commands and Data
3	VSS1	S	Supply Voltage Ground
4	VDD	S	Supply Voltage
5	CLK	I	Clock
6	VSS2	S	Supply Voltage Ground
7	DataOut	O	Card to Host Data and Status
8	RSV ⁽²⁾	I	Reserved
9	RSV ⁽²⁾	I	Reserved

NOTES: 1) S=power supply; I=input; O=output.

2) The 'RSV' pins are floating inputs. It is the responsibility of the host designer to connect external pullup resistors to those lines. Otherwise non-expected high current consumption may occur due to the floating inputs.

Şekil 4.3 SPI modundaki pin bağlantıları ve isimleri

