White Paper

Reducing CompactFlash Re-Qualification Time

Introduction

Industrial grade CompactFlash (CF) has been a storage workhorse for embedded systems since the turn of the century, and it appears as though this tried and true storage platform still has a long life ahead. In a market where, "if it ain't broke, don't fix it" still rings true, CF provides the boot and data logging functions for a multitude of infrastructure applications from radio base stations to routers and switches, to transportation monitoring systems.

But while many OEMs still want to use industrial CF, industry consolidation and NAND transitions have forced infrastructure system OEMs into yet another re-qualification period. Many of these systems could have been designed over 10 years ago and sustaining engineers must deal with the fact that the original engineering teams may not be available to deal with any incompatibilities that may occur. With all the permutations and combinations of operating modes that are available in the market, where does an engineer start in terms of re-qualifying a new CF card into a current system?

Identify Device Information

Figure 1 shows a partial Identify Device Information table from a Virtium TuffDrive® CF datasheet. Most industrial CF manufacturers will have a similar table that defines the "personality" of the CF card.

Word	Description	Default Value
0	True IDE mode PCMCIA mode	045Ah 848Ah
1	Number of cylinders	XXXXh
2	Reserved	0000h
3	Number of heads	00XXh
4	Number of unformatted bytes per track	0000h
5	Number of unformatted bytes per sector	XXXXh
6	Number of sectors per track	00XXh
7-8	Number of sectors per card	XXXXh
9	Reserved	0000h
10-19	Serial number (20 ASCII characters)	XXXXh
20	Buffer type:	0002h

Word	Description	Default Value
	0000h: Not specified 0001h: A single-ported, single-sector buffer 0002h: A dual-ported multisector buffer 0003h: A dual-ported multisector buffer with a read caching	
21	Buffer size in 512-byte increments	0002h
22	Number of ECC bytes passed on read/write long commands	0004h
23-26	Firmware revision (8 ASCII characters)	XXXXh
27-46	Model number (40 ASCII characters)	XXXXh

Figure 1: Identify Device Information

This personality, however, depends on the electrical connection between the host and the CF. For example, if the host operates the CF in true IDE mode (pin 9, -ATA SEL grounded), word address 0 will read 045Ah. If in PCMCIA mode, word address 0 will read 848Ah. There are several other fields that may change based upon the operating mode as well.

Ensuring Compatibility

The single best way to ensure compatibility is to take a snapshot of the identify device information from a known working CF when it is in the target system using the Identify Device (0xEC) command. Providing this 512 bytes of data to a new CF vendor will virtually ensure compatibility, unless the application software was written to specifically look for a vendor-specific ID string or something similar. The example below shows how to retrieve the identify device information in a Linux-based system.

From the command line (do not type the brackets { }):

```
> sudo –s                                                   ‘grants administrative rights
> lshw –short –c disk                                       ‘lists all the drives in the system
> hdparm –I {device mounting point}                         ‘see note 1
> hdparm –I Istdout {device mounting point} > {filename}.bin  ‘see note 2
```

Notes:

1. This command outputs most of the identify device information in a human readable format, but it does not output everything.
    a. Device mounting point = the path from the lshw command. For example: lshw –short –c disk might output a device mounting point of /dev/sdb
    b. Then the command would be > hdparm –I /dev/sdb
2. This command retrieves the binary data of all 512 bytes of the identify device information and exports it to a file that can be e-mailed. Virtium application engineering can then interpret that file and suggest the proper product configuration.

While in-system identify device information is best, it is not always easy or practical to get this information without some help from the system software developers. The next best thing is to connect a known-working CF to a Windows based system and run an off-the-shelf utility like Passmark’s Diskcheckup to read the identify device information. While the operating mode might not be identical, there is usually enough beneficial information to “get close.”

If neither of these options is viable, the only other alternative is to try to get as much information about the CF and the system as possible.

Key Parameters

At a minimum, it is important to understand the operating mode of the CF (or at least the modes of operation a known-working card supports). The vast majority of compatibility issues could be avoided if the operating mode was fully understood up front. Sometimes this information can be found in the datasheet:

Fixed IDE or autosense?

- Autosense means that the CF identifies itself as a fixed device in True IDE mode (pin 9 grounded) or, if operated in PCMCIA mode, as a removable device

CHS (cylinder, head, secotor) and/or LBA (logical block address) values?

- Regardless of which mode is being used, it is generally important to know the CHS and LBA values of a known working card. Many software engineers used CHS or LBA values as a check to make sure that a qualified CF is being used.
- A new CF card that has a lower LBA count than a currently qualified card might have issues with duplication in production.

Compliance

- CFA (ex. CFA 3.0, 4.1, 6.0)
- PCMCIA (ex. 2.1)
- ATA (ex. ATA-8, ATA-3)
  - Many older systems were designed around the ATA-3 standard. Many new CF cards are at least ATA-6 compatible. Several ATA-3 words in the identify device information became “don’t cares” in subsequent ATA revisions. If host system software considered these words important, then incompatibilities could arise.

UDMA and/or MWDMA supported and if so, what level?

- Some older systems may have difficulties handshaking with newer CF cards. CF vendors can cap modes to make that handshaking easier.
  - For example, even though TuffDrive CF could support UDMA mode 6, Virtium has decided to cap the CF to UDMA mode 4 for best general compatibility.

Conclusion

CompactFlash has proven to be a highly reliable storage device for industrial embedded systems. Industry consolidation has seen several CF manufacturers exit the business. Even those that remain continue to have to deal with NAND flash obsolescence, forcing OEMs to re-qualify new CompactFlash technology. Most systems into which CF has been designed have been deployed for many years and their original design teams may be on to other products. This leaves sustaining engineers with the burden of having to qualify new CF with very few resources. Understanding key parameters such as CHS/LBA values and operating modes can greatly reduce the qualification burden.

Virtium plans to continue to offer its TuffDrive CF products at least through the end of the decade and can offer tools and application engineering support to make re-qualification easier.

For more information, please contact your Virtium representative, e-mail us at CF@virtium.com, or visit our website at http://www.virtium.com.

Virtium manufactures memory and storage solutions for the world's top industrial embedded OEMs. For 18 years we have designed, built and supported our products in the USA - fortified by a network of global locations. Our world-class technology and unsurpassed support provide a superior customer experience that continuously results in better industrial embedded products for our increasingly interconnected world.

Virtium
Memory and Storage Solutions

30052 Tomas | Rancho Santa Margarita, CA 92688
Phone: 949-888-2444 | Fax: 949-888-2445
www.virtium.com

WP004-0115-01