

CSM FAT File System



- ❑ Unterstützung der von MS-DOS bekannten Eigenschaften:
 - 12- und 16-Bit-FATs
 - Unterverzeichnisse
 - Dateinamen im 8.3-Format
 - Dateiattribute, -datum, -uhrzeit
- ❑ Portierbar durch Quelltext ANSI C
- ❑ Für 8-, 16- und 32-Bit Prozessoren und Microcontroller
- ❑ Zahlendarstellung im INTEL- oder MOTOROLA-Format
- ❑ Sparsame Verwendung von Speicher
- ❑ Geeignet für ATA PC Cards, CompactFlash Cards, MultiMediaCards, SD Memory Cards und SRAM Cards

Einführung

Das von Microsoft mit **MS-DOS** eingeführte Dateisystem hat eine große Verbreitung erlangt. Daher bietet sich dieses Dateisystem auch für die Datenspeicherung auf kundenspezifischen Systemen sowie zum Austausch zwischen diesen Systemen und PCs an.

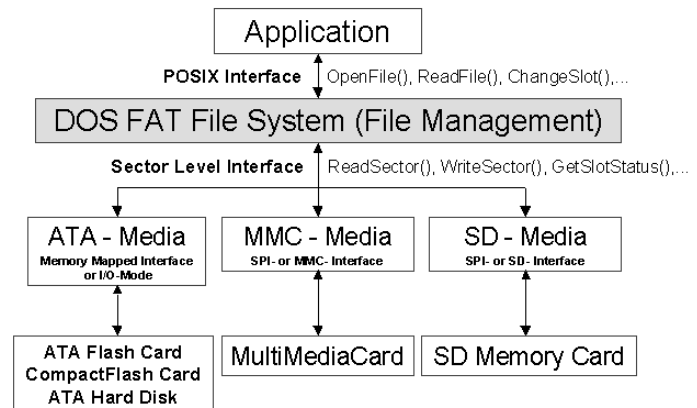
Als Datenträger können **ATA Cards, CompactFlash Cards, MultiMediaCards, SD Memory Cards** oder **SRAM Cards** verwendet werden.

Neben der Hauptanwendung muss die Software des Zielsystems zusätzlich Funktionen zur Verwaltung des Dateisystems (*file system*) und Funktionen zum Ansprechen der Memory Cards (Low-Level Funktionen) enthalten.

Bei der Implementierung des Dateisystems kann der Entwickler auf das **CSM FAT File System** zurückgreifen. In diesem Entwicklungskit enthalten ist unter anderem ein portierbarer, dokumentierter **ANSI C Quelltext**, der ein zu MS-DOS V6.x kompatibles Dateiverwaltungssystem realisiert. Damit können PC Cards direkt von **Windows Betriebssystemen** gelesen und geschrieben werden.

Der Entwickler kann mit **CSM FAT File System** ein bewährtes und getestetes Modul in seine Software übernehmen.

Software Architecture



Development Kit

Mit dem **CSM FAT File System – Development Kit** wird ein lauffähiges Gesamtsystem für folgende Standard-Plattform geliefert:

- **Standard PC mit Centronics Interface**
- **Betriebssystem MS-DOS V6.x, WIN 95/98/Me oder Win NT/2000**
- **Memory Card Laufwerk CSM OmniDrive Professional**
- **Entwicklungsumgebung Microsoft Visual C++ 6.0 (WIN32) oder Borland C V3.1(16-Bit)**

Für diese Plattform sind die Low-Level Funktionen zum Zugriff auf die Memory Card im OmniDrive als Modul enthalten.

Einige **Beispielanwendungen** mit Quelltexten und Konfigurationsdateien für die Compilierung runden das Paket ab.

Das **CSM FAT File System** wird als Quelltext-Modul in die Beispielanwendungen eingebunden.

Für das **CSM FAT File System** wird eine **Projektlizenz** erteilt, d.h. von dem vereinbarten Projekt dürfen beliebige Stückzahlen ohne weitere Lizenzabgaben hergestellt werden. CSM erwartet in diesem Zusammenhang die Unterzeichnung einer **Geheimhaltungserklärung** zum Schutz der Interessen beider Seiten.



Eigenschaften

Komfortable, datei-orientierte **Schnittstelle zur Anwendung** (in Anlehnung an POSIX):

- Erzeugen, Öffnen und Schließen von Dateien: CreateFile(), OpenFile(), CloseFile()
- Lesen und Schreiben von Daten: ReadFile(), WriteFile()
- Positionieren in einer Datei: SeekFile(), FlushFile()
- Umbenennen und Löschen von Dateien: RenameFile(), DeleteFile()
- Bearbeiten von Dateiattributen: GetFileMode(), SetFileMode(), GetFileDateAndTime()
- Verzeichnis anlegen, wechseln und löschen: MakeDir(), ChangeDir(), RemoveDir()
- Verzeichnis durchsuchen: ReadDir()
- Auswahl des Laufwerks, Formatieren des Datenträgers: ChangeSlot(), FormatVolume()

Hohe **Portierbarkeit** (für den Einsatz auf Microcontrollern geeignet):

- **Byte Order** (Big-Endian oder Little-Endian) per bedingter Compilierung auswählbar

Verwendung

CSM FAT File System ist prinzipiell für eine sehr große Bandbreite von Systemen geeignet:

- **8-Bit-, 16-Bit- und 32-Bit-Prozessoren und Microcontroller**
 - ✓ 8051 und Derivate
 - ✓ Intel x86 (8088, 8086, 286, 386, 486, Pentium) und kompatibel
 - ✓ Siemens C16x-Familie
 - ✓ Motorola 680x0, 683xx und weitere
 - ✓ ...
- **sparsame Verwendung von Speicher**
 - ✓ ca. 4,5 KB RAM
 - ✓ ca. 5 KB bis 10 KB Codespeicher
- diverse PC Card **Hardwareanbindungen**
 - ✓ PC Card Controller
 - ✓ im Adressraum
 - ✓ im I/O-Bereich
 - ✓ PC Card Laufwerk
 - ✓ ...
- verschiedene **PC-Card Typen**
 - ✓ ATA Schnittstelle (ATA Flash Card, CompactFlash Card, Hard Disk)
 - ✓ SPI- oder MMC-Interface (MultiMediaCard)
 - ✓ SPI- oder SD-Interface (SD Memory Card)
 - ✓ Lineare Adressierung (SRAM Card)

		CSM FAT File System Development Kit		System des Anwenders	
Plattform:	Standard PC mit Centronics Interface, MS-DOS V6.x, WIN 95/98/Me/ NT/2000, Microsoft Visual C++ 6.0 (WIN32) oder Borland C V3.1(16-Bit)			projekt-spezifische Hardware	
	Beschreibung	Quell	EXE	Beschreibung	Tätigkeit
Anwendung:	Beispiel-Anwendungen (_COPY, _FORMAT)	✓	✓	kundenspezifische Anwendungssoftware des System	in der Verantwortung des Entwicklers
Schnittstelle:	datei-orientiert ("POSIX")			datei-orientiert ("POSIX")	
CSM FAT File System:	gelieferter Quelltext (ohne Änderungen)	✓	✓	portierter Quelltext	ggf. Anpassung des Quelltextes an die Hardware-Plattform
Schnittstelle:	sektor-orientiert			sektor-orientiert	
Low-Level-Funktionen:	Modul für CSM OmniDrive Professional		✓	Low-Level-Funktionen Zugriffsroutinen	nicht im Lieferumfang, vom Entwickler erstellt
Hardware:	CSM OmniDrive Professional			projekt-spezifisch	

Aufbau

Die Anwendung des **CSM FAT File System** erfolgt in einem Drei-Schichten-Modell:

- Anwendung (obere Schicht)
- **CSM FAT File System** (mittlere Schicht)
- Low-Level-Funktionen (untere Schicht)

CSM FAT File System – Development Kit enthält für die oben genannte Standard-Plattform eine lauffähige Referenz-Realisierung:

- Beispielanwendungen als C-Quelltext und ausführbar als EXE: `_COPY`, `_FORMAT`
- **CSM FAT File System** als ANSI-C-Quelltext, auf Standard-Plattform ohne Änderung einsetzbar
- Low-Level-Funktionen als Modul für CSM OmniDrive Professional (Quelltext nicht erforderlich)

Für den Einsatz **CSM FAT File System** auf einer anderen Hardware-Plattform (z.B. Microcontroller-system, Steuerung, Industriecomputer etc.) integriert der Entwickler den Quelltext des **CSM FAT File System** in seine Software. Diese kann folgende Struktur haben:

- kundenspezifische Anwendungssoftware des Systems
- **CSM FAT File System**, vom Entwickler auf die verwendete Plattform portiert
- Low-Level-Funktionen mit sektor-orientierter Schnittstelle und Zugriffsroutinen auf Memory Card, wird vom Entwickler für die verwendete Plattform erstellt

CSM FAT File System benötigt wenige, genau spezifizierte Low-Level-Funktionen:

- Lesen/Schreiben eines 512-Byte-Sektors: `ReadSector()`, `WriteSector()`
- PC-Card gesteckt? Card gewechselt?, Card schreibgeschützt?: `GetSlotStatus()`
- Speicherkapazität der Karte: `GetSizeOfCard()`
- Datum und Uhrzeit (für Dateidatum und -uhrzeit): `DOSDate()`, `DOSTime()`

Portierung

Bei der **Entwicklung der Software** für die neue Hardware-Plattform (z.B. embedded system mit Microcontroller) finden unter anderem folgende Tätigkeiten statt:

➤ **Bereitstellen von Zugriffsroutinen auf die Memory Card.**

Diese Routinen sind von der neuen Hardware abhängig. Hier muss auch entschieden werden, welche Typen von Memory-Cards zum Einsatz kommen sollen (ATA Cards, CompactFlash Cards, MultiMediaCards, SD Memory Cards oder SRAM Cards).

Für diese Routinen gibt es von Seiten des **CSM FAT File System** keine Vorgaben. Es sollten jedoch die Empfehlungen der jeweiligen Standards (z.B. PC Card Standard, MultiMediaCard Standard) beachtet werden.

➤ **Realisierung der Low-Level-Funktionen gemäß der Schnittstellen-Definition des CSM FAT File System.**

Diese Funktionen bedienen sich sinnvollerweise der Zugriffsroutinen (siehe voriger Punkt). Mit einer sektor-orientierten Schnittstelle kapseln die Low-Level-Funktionen alle hardware- und Memory-Card-abhängigen Details der kundenspezifischen Plattform zum **CSM FAT File System** hin.

➤ **Portierung des CSM FAT File System.**

Der Quelltext ist auf bestmögliche Portabilität ausgelegt. Bei der Vielzahl möglicher Hardware-Plattformen und System-umgebungen sind Anpassungen jedoch nicht völlig auszuschließen. Die mitgelieferte Dokumentation unterstützt den Entwickler dabei so weit wie möglich. Folgende Punkte verdienen Beachtung:

➤ **Ausrichtung (alignment) der Daten:**

Auf dem Datenträger liegt das Speicherformat fest, das sich auch in einigen Datenstrukturen **CSM FAT File System** widerspiegeln muss.

➤ **Zugriff auf Datenstrukturen:**

Bei Systemen, die nur auf Datenobjekte auf bestimmten Adressen zugreifen können, muss auf korrekte Adresslage und Zugriffe geachtet werden.

➤ **Multitasking-Umgebung:**

Es muss im Einzelfall geprüft werden, ob das **CSM FAT File System** unverändert eingesetzt werden kann. Gegebenenfalls müssen geeignete Verriegelungsmechanismen zwischen konkurrierenden Prozessen verwendet werden.

- **Echtzeit-Umgebung:**
Die Eignung des **CSM FAT File System** für bestimmte Anforderungen an das Zeitverhalten muss für die spezielle Anwendung untersucht werden.
- **Integration mit Applikation:**
Verwendung der von **CSM FAT File System** bereitgestellten datei-orientierten Schnittstelle durch die Anwendung.

Die **Integration** des **CSM FAT File System** in die kundenspezifische Gesamtsoftware für eine bestimmte Hardwareplattform erfordert - wie jede verantwortungsbewusste Entwicklungstätigkeit - ein gewisses Maß an **Fachkenntnis**.

CSM FAT File System - Development Kit gibt dem Entwickler wertvolles Wissen über das Dateisystem in Form von Dokumentation und Quelltext (Programmcode und Kommentar).

Lieferumfang CSM FAT File System:

- **CSM FAT File System** als ANSI-C-Quelltexte (bestehend aus den Dateien `File.c`, `File.h` und `MyDefs.h`).
Die Quelltexte sind ausführlich und sinnvoll kommentiert.
- **Dokumentation** (Referenzhandbuch): Überblick, Beschreibung der gelieferten Dateien, Referenz der zur Verfügung gestellten Funktionen, Variablen und Datenstrukturen, Hinweise für die Anpassung an verschiedene Konfigurationen und Prozessoren bzw. Microcontroller
- **Lauffähiges Modul** (untere Schicht) für das Memory Card-Laufwerk CSM OmniDrive auf der Standard-Plattform.
- **Beispielanwendungen** in Form von C-Quelltexten.
- **Projekt- und Konfigurationsdateien** speziell für Microsoft Visual C++ 6.0 (WIN32) oder Borland C ab V3.1(16-Bit)
- **Halbtägige Applikations-Schulung bei CSM** Anwenderspezifische Schulung, Informationen zur Anwendung **DOS FAT File System**, Hinweise für das Design-In des Speichermediums

Applikations-Schulung

Der Erwerb des **CSM FAT File System** beinhaltet eine halbtägige Schulung für einen Mitarbeiter, bei der wichtige Informationen zur Anwendung des **DOS FAT File System** sowie zum Interface für das verwendete Speichermedium zur Verfügung gestellt werden. Die Schulung erfolgt jeweils individuell für den Entwickler und kann nach Absprache auf die geplante Applikation abgestimmt werden (z.B. Typ des Microcontrollers).

Die Schulung findet bei CSM Filderstadt statt.

CSM GmbH, Raiffeisenstr. 34, D-70794 Filderstadt

Tel.: +49 711 77964 20 Fax: +49 711 77964 40

mailto: info@csm.de http://www.csm.de

Alle erwähnten Marken- oder Warenzeichen sind Eigentum ihrer jeweiligen Besitzer. Technische Änderungen vorbehalten.