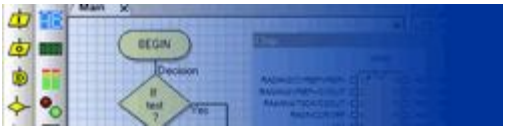
Introduction

The Matrix Multimedia FAT16 Flowcode component is designed to be used in conjunction with the Matrix Multimedia E-Block MMC / SD board to provide an introduction to using and configuring the FAT file system located on modern flash memory card technology. The component will however work with any SD / MMC card that is formatted to the FAT16 file structure using the SPI bus protocol.

The FAT16 component has the following Macros and Properties:
Macros
Properties
Connections

Currently Unsupported in the FAT16 component.
+ Long file names
+ FAT32 file system
+ Deleting data from a file
+ Deleting directorys
+ Directories other then root cannot have more then 512 entries including additional long file name entries.



Macros

The FAT16 component has quite a lot of macros as a large number of possible functions are needed for FAT file system interrogation. This section will list the macros, their parameters, and their associated functions.

char Init_Fat (void)

Initialises the card and starts up the FAT driver, points the directory to root.
Current directory is specified as Root.

Retval = 0 - Init OK
Retval > 0 - Init Not OK

char Open_File("filename")

Searches for a file in the current directory and reads the first sector of the file into memory.

Prerequisites - Init_FAT
Parameters - Filename string

Retval = 0 - File found and opened - 512-byte buffer contains 1st sector of file
Retval = 1 - File not found
Retval > 1 - Error

Read_File_Sector()

Overwrites local buffer with the current file sector from the disk - Used to restore the local sector buffer with the actual sector stored on the card.

Prerequisites - Init_FAT, Open_File
Optional - Move_To_Next_Sector

Retval = 0 - Local buffer refreshed with current card sector
Retval > 0 - Error

Read_File_Length()

Returns the number of bytes in the current sector that are used by the current open file.

Prerequisites - Init_FAT, Open_File

Retval = 512 - Sector is filled with file data
Retval < 512 - Last sector of the file, Retval specifies how many bytes of data are remaining in the sector

Move_To_Next_File_Sector(Force_Next_Sector **)**

Overwrites local buffer with the next file sector from the disk - Used to read the next chunk of a file. If the force next sector parameter is set to 0 then when the last sector is reached the function will remain in the last valid sector of the file. If the force next sector parameter is set to 1 then when the last sector is reached the function will create a new sector at the end of the file and update the size

parameter of the file.

Prerequisites - Init_FAT, Open_File
Parameters - Force_Next_Sector

Retval = 0 - Sector read or appended correctly
Retval > 0 - Error

Append_String_To_File("String")

Appends a string or byte array of data to the end of the file that is currently loaded and updates the size parameter of the file.

Prerequisites - Init_FAT, Open_File
Parameters - String - Contains the String or byte array to be appended to the end of the file

Retval = 0 - String or byte array has been appended correctly
Retval > 0 - An error occured while appending the string.

Write_File_Sector()

Overwrites the current sector on the memory card with the contents of the local sector buffer. See write_byte_to_buffer function.

Prerequisites - Init_FAT, Open_File, Write_Byte_To_Buffer
Optional - Move_To_Next_File_Sector
Parameters - String - Contains the String or byte array to be appended to the end of the file

Retval = 0 - File sector written successfully.
Retval > 0 - File sector write error

Create_File("String")

Creates a file with the specified filename in the current directory of the card. Also assigns the file certain parameters such as create time and date and size etc.

Prerequisites - Init_FAT
Optional - Open_Folder
Parameters - String - Contains the String or byte array to be appended to the end of the file

Retval = 0 - File created successfully
Retval > 0 - Directory table of the current folder is full.

Delete_File("String")

Scans the current directory on the card for a specific filename. If the filename is found then the file is removed from the disk by destroying the FAT entries in the directory entry.

Prerequisites - Init_FAT
Optional - Open_Folder
Parameters - String - Contains the name of the file to be deleted.

Retval = 0 - File found and deleted.
Retval > 0 - File not found

Write_Byte_To_Buffer(Address, Data)

Modifies the value of a byte in the local sector buffer.

Prerequisites - Init_FAT, Open_File
Optional - Move_To_Next_File_Sector, Read_File_Sector
Parameters - Address = 0 - 511 - Contains the position of the byte to be updated.
Parameters - Data - Contains the byte value that will be written to the local sector

Read_Byte_From_Buffer(Address)

Reads a byte of data from the local sector buffer.

Prerequisites - Init_FAT, Open_File
Optional - Move_To_Next_File_Sector, Read_File_Sector
Parameters - Address = 0 - 511 - Contains the position of the byte to be read.

Retval = Data byte read from the local sector buffer.

Open_Folder("String")

Scans the current directory on the card for a specific folder name. If the folder name is found then the card opens the folder and makes it the current directory.

Prerequisites - Init_FAT
Parameters - String - Contains the name of the folder to be opened.

String = "root" - Go directly to Root folder
String = ".." - Go up a directory

Retval = 0 - Folder found and opened
Retval > 0 - Folder not found

Scan_Current_Folder(Current_Idx, Scan_Type)

Method for retreiving file or folder names for the card to allow for dynamic file opening or playback.

Prerequisites - Init_FAT
Optional - Open_Folder
Parameters - Current_Idx - Contains the current pointer in the directory table.
Parameters - Scan_Type - Sets the scan to search for Files or Folders.

Current_Idx - Call the function with values from 0 to 511 to completley search the current folder for a files or folders
Scan_Type - 0 = Scan for files, 1 = Scan for folders

Retval = 0 - File or folder found and result stored into string array.
Retval = 1 - End of folder reached.
Retval = 2 - Filetype in the directory table does not match the Scan_Type parameter.
Retval = 3 - Current section of the Filetable is used for an extended filename or is empty.

Read_Byte_From_Scan(Idx)

Returns a byte of the name of the file or folder retreived from the Scan_Current_folder function.

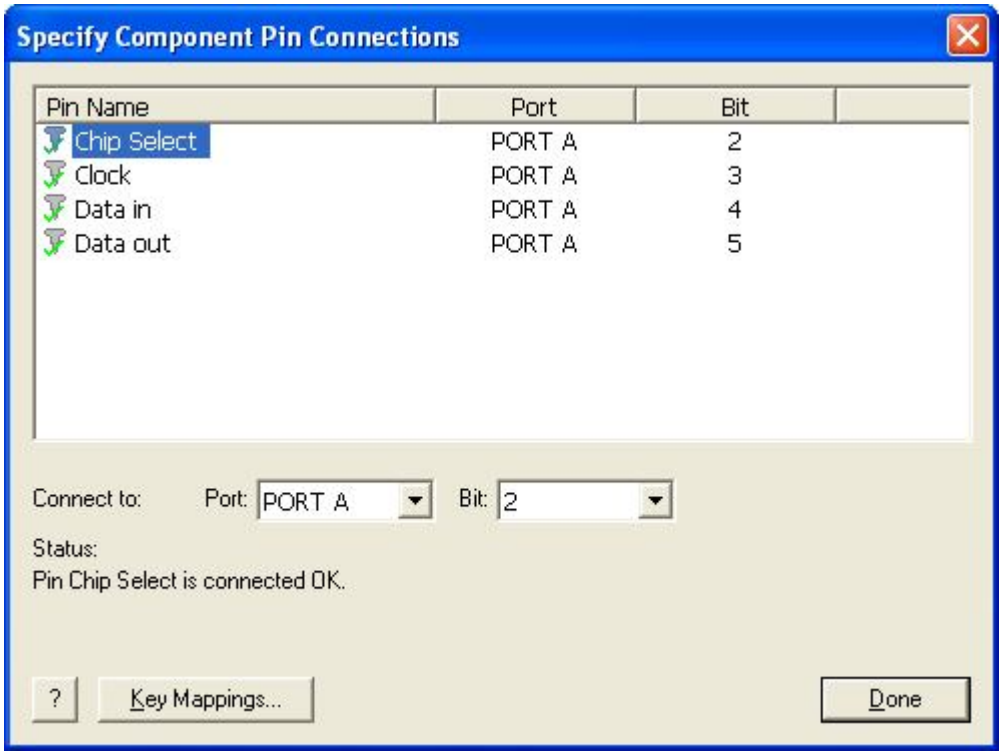
Prerequisites - Init_FAT, Scan_Current_Folder
Parameters - Idx - Contains the current pointer in the file or folder name retreived from the last scan.

Idx - 0 - 11 for a file returns the 8.3 character file name.
Idx - 0 - 7 for a folder returns the 8 character folder name#

Retval = ASCII byte at position Idx from the folder name or file name at position Current_Idx in the directory

Connections

Component Connections
The FAT16 component has the following connections.



Note: If the FAT16 component is not using hardware SPI then the SPI bus is bit banged therefore any I/O pins can be used. If the FAT16 component is using hardware SPI then all of the pins except the chip select pin are defined by the chips peripheral SPI pins (SDI, SDO, SCK).