

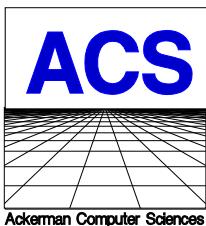
Compact FLASH II

Digital Audio

System

Revision 2.23

Tuesday, May 16, 2006



On The Cutting Edge of Technological Evolution

6233 E. Sawgrass Rd • Sarasota, FL 34240 • (941)377-5775 FAX(941)378-4226
www.acscontrol.com



Copyright(c)1992-2006 by ACS, Sarasota, FL ALL RIGHTS RESERVED

Table of Contents

Overview.....	3
Features.....	3
Connections	4
Main Connector.....	4
Serial Connector	5
Programming the Compact FLASH Card	6
Installing and Removing the Compact FLASH Card	6
Audio Rates Supported.....	6
Filename Formats	7
Sample Filenames.....	8
Compact FLASH Card Loading and Scanning	9
Background Audio Tracks.....	9
CFSOUND.INI File.....	10
Quiz / Kiosk Mode	12
Sequential Sound Playout.....	14
RS-232 Serial Port Protocol	15
Card Requirements	16
Diagnostic LED Codes (Red & Green LEDs).....	16
Green LED.....	16
Red LED Error Flashes.....	16
Power Requirements	16
Digital Audio Recording	17
Recommended Recording Procedure	17
Optional Plug-in Modules.....	18
CFSound II Contact Sense 8 Module	18
CFSound II Contact I/O 8 Module	19
CFSound II Contact Sense 24 Module	21
Contact Wiring	22
Frequently Asked Questions	23
Sample Messages	27
Sample Power Up Message	27
Sample Card Insertion Messages.....	27
SanDisk 8meg.....	27
Dane-Elec 16meg	28
Kingston 8meg	28
Memorex 8meg.....	28
Sample Directory Listing.....	29
Mechanical.....	30
Firmware Revisions	31

Overview

Thank you for purchasing the **Compact FLASH II Digital Audio System**; be assured that it will provide you with thousands of hours of trouble free, solid-state operation. You will also find that it is simple to change audio themes or tracks and is easy to make your own recordings or “program” the device.

End users can change the audio themes or tracks simply by inserting another, different Compact FLASH Card, as easily as the changing of a cassette tape. Multiple cards can be programmed with different audio themes or tracks allowing daily, hourly, weekly or monthly changes. It truly is as simple as removing the old card and plugging in the new one. Depending on the manufacturer, Compact FLASH Cards may be re-used up to 10,000 times.

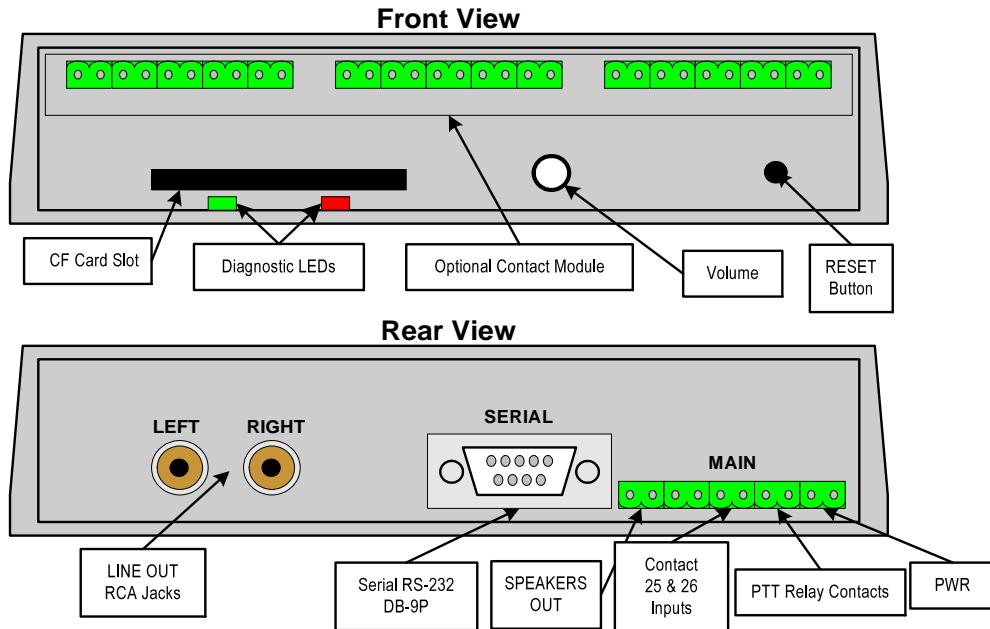
Sound files are easily copied onto the card via your PC or laptop, using a Compact FLASH to PCMCIA, Compact FLASH to USB or parallel port adapter. Once the files are copied to the Compact FLASH Card, they are renamed. This process instructs the **Compact FLASH II Digital Audio System** how and when to play the copied files.

Features

The Compact FLASH II Digital Audio System incorporates the following features:

- Uses inexpensive, industry standard Compact FLASH (CF) Cards up to 512meg in size.
- A 32meg CF Card provides up to 69 minutes of audio at an 8KHz sampling rate and up to 34 minutes at a 16KHz sampling rate.
- RS-232 Serial Port for controlling audio play out via an attached computer or PLC.
- Diagnostic LED's to indicate operating status.
- Built-in dual contact input.
- Push To Talk (PTT) relay contact that can optionally close whenever a sound is played.
- Built-in Stereo Amplifier for 4 or 8 ohm speakers.
- Optional board for contact inputs to activate sounds.
- Optional board for contact outputs activated with sound for sound steering or other control.

Connections



Main Connector

Most of the connections to the CFSound-II are supplied to the unit with a ten pin, 2 piece pluggable terminal block style connector **Main**.

Power Supply Connection

The CFSound II requires a regulated or unregulated voltage of 10V to 18V DC. The unit has resettable fusing internally to protect the Power input. Should the fuse trip, remove power, wait a moment for the fuse to reset, then re-apply power. The non-OEM version of the CFSound II is supplied with a wall mounted transformer rated at 15VDC @ 0.8A.

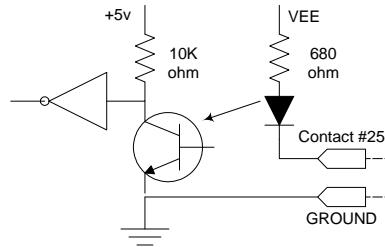
ACS supplies two different versions of the wall-mounted transformer power supply for the CFSound-II. The version in the black enclosure has a white stripe on the positive lead which should be connected to pin 9 of the **Main** connector with the other lead to pin 10. The version in the beige enclosure has a black stripe on the negative lead which should be connected to pin 10 of the **Main** connector with the other lead to pin 9.

Built-in Contacts Connection

The Push-To-Talk (PTT) contact output on the **Main** connector may be activated whenever a sound is playing. The contact output is implemented with a relay whose contacts are rated at 1A @ 30VDC. The common and Normally Open relay contacts are available on the **Main** connector.

The contact inputs on the **Main** connector appear logically to the CFSound-II as Contact #25 & #26, and may be activated by momentarily applying ground to the Input Contact #xx pin on the **Main** connector. The contact inputs are implemented with an optocoupler. The Cathodes of the LEDs in the optocoupler are connected to the Contact #25 and #26 Input pins. The Anodes of the LEDs in the optocoupler are connected to an internal 12VDC power supply, with a 680 ohm current limiting resistor in series. The output transistor of the optocouplers have 10K pullup resistors on their collectors, with the

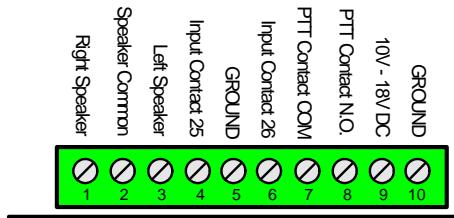
emitters connected to ground and are buffered by an inverting gate. The following diagram is representative of the Contact #25 input and is similar for Contact #26:



The input is activated by sinking current from the Contact input pin to ground. A Ground connection is supplied on pin 5 of the **Main** connector for this purpose. The input current sink requirement is approximately 15 mA.

Speaker Connection

The speaker terminals on the **Main** connector provide connections for the two external left and right speakers. These may be either 4 or 8 ohm speakers, with 4 ohms providing higher output power. The speakers should be capable of handling 4 watts.



Pin #	Signal
1	Right Speaker
2	Speaker Common
3	Left Speaker
4	Input Contact 25
5	Ground
6	Input Contact 26
7	PTT Contact COM
8	PTT Contact N.O.
9	10VDC – 18VDC
10	Ground

Serial Connector

A standard RS-232 serial interface is provided on-board via the DB-9 style male **Serial** connector. The CFSound II provides diagnostic messages about CF card insertion and removal events and sound contents over the serial port. Additionally, the playing of sounds may be controlled by messages issued to the CFSound II via the serial port. The format of the serial data is 2400 baud, 8 data bits and 1 stop bit. Only three of the nine pins are used. The configuration of jumper block JB2 on the CFSound II controls whether the serial port looks like a Modem (DCE) or a Terminal (DTE).

As shipped, the CFSound II is configured to be DCE so that a one-to-one cable from a PC serial port may be used. The pinout for the **Serial** connector is:

Serial Pin #	Signal Name JB2 = DCE	Signal Name JB2 = DTE
2	TX (from unit)	RX (to unit)
3	RX (to unit)	TX (from unit)
5	Ground(GND)	Ground(GND)

Programming the Compact FLASH Card

The Compact FLASH Card is programmed by inserting it into the PCMCIA slot of a Compact FLASH to PCMCIA adapter or a Compact FLASH adapter connected to your parallel port. Sound files are copied to the Compact FLASH Card just as if it were a disk drive. The files used are the standard “*.WAV” files typically employed by Windows and DOS audio applications for years.

When a new Compact FLASH Card is first installed into your computer, Windows (95 or later) should recognize the device and install the software necessary to access the card. After the software installation is complete, the Compact FLASH Card appears as a disk drive icon under My Computer. You may then use Windows Explorer to drag and drop the sound files from your computer’s hard disk directly to the Compact FLASH Card icon. It really is just that easy.

Please take care not to use audio tracks that are copyrighted. If you wish to avoid this issue, you will find that many computer software outlets have CD’s with non-copyrighted music and audio sound effects that you may use without limitation. They are commonly found in the multimedia section. Please be advised that you as the user are solely responsible for audio that you record and play from the **Compact FLASH II Digital Audio System**.

Installing and Removing the Compact FLASH Card

Care should be taken when installing or removing the Compact FLASH Card. Please be certain to insert the card with a straight and level motion. Please do not force the card to insert. Otherwise, some of the 50 pins in the socket may become damaged and the card and/or **Compact FLASH II Digital Audio System** may become unusable.

Before REMOVING the Compact FLASH Card under Windows 95 or later, Microsoft recommends that you double click on the PCMCIA icon at the bottom of your computer screen. This will bring up the pc card [PCMCIA] properties window. Click on socket status tab, then select the card you wish to remove by clicking on it. Finally, click on the stop button. Windows will then tell you that the card socket services have been stopped.

On the **Compact FLASH II Digital Audio System** this will all happen automatically simply by removing the card, although its not recommended that the card be inserted or removed while power is applied.

Audio Rates Supported

The Compact FLASH II Digital Audio System supports mono 8-bit Wave files at a sampling rate of up to 22KHz (22,050Hz) and stereo 8-bit Wave files at a sampling rate of up to 11KHz (11,025Hz). Lower sampling rates are supported for longer play-outs. Monaural Wave files are played on both channels. The **Compact FLASH II Digital Audio System** reads the sampling rates and formats encoded in the .WAV sound files whenever a card is first inserted. The **Compact FLASH II Digital Audio System** will not play any file that does not meet these requirements. Further, this error condition should be identified by the red led flashing twice every six seconds.

NOTE: If high sample rate audio files play-out with skips, repeats or stop playing before completion, you may need to re-format the Compact FLASH Card to eliminate the condition known as excessive file

fragmentation. Compact FLASH cards must only be formatted using the FAT12 (floppy) or FAT16 file system. FAT32 or NTFS file system formats are not supported. The maximum supported partition size is 32meg. (512meg with version 2.1 or higher firmware).

Filename Formats

The filenames of the individual Sound files on the Compact FLASH Card control how and when the sounds are played out by the **Compact FLASH II Digital Audio System**. The filenames must start with the optional input number they are associated with expressed as a two digit HEX number.

The HEX numbering system has a base of 16 as compared to the decimal system which has a base of ten. A HEX numeric digit is 0 through F, 0 being equal to decimal 0, and F being equal to 15. The first digit (MSB) of the two digit hex number is multiplied by 16. The second digit (LSB) is added to the adjusted value derived from the first digit. The number 10 in HEX is thus equal to 16 decimal. The number 20 in HEX is 32 decimal, 21 in HEX is 33 decimal, and 7F in HEX is 127 decimal. For your convenience, a decimal to hex conversion table is provided below:

LSB → MSB↓	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
2	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
3	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
4	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
5	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
6	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
9	144	145														

For example, a sound file associated with optional contact input #1 would be named “01.WAV,” a sound file for input #10 would be named “0A.WAV,” a file for input #16 would be “10.WAV” and a file for input #36 would be “24.WAV”. There is no input #0 and so “00.WAV” is reserved and should not be used.

The two digit hex input number must be the first two characters in the filename.

Additional letter tokens are appended to the input # in the filename to determine *when* and *how* the audio sound is played. The **Compact FLASH II Digital Audio System** provides one input in its standard configuration and, by adding an optional board, can be expanded to either 9 or 25 total inputs. The inputs can be wired to switches or other such devices, either normally opened or normally closed.

You can also connect the RS-232 port of the **Compact FLASH II Digital Audio System** to the RS-232 port of your computer or PLC to control up to 145 sounds using serial commands.

The filename letter tokens are described as follows:

If a “**C**” appears after the hex number in the filename, the sound file will be played on a *closure* of the corresponding optional contact input module input or serial command. The sound will start playing when the associated contact input goes from an open to closed state.

If a “**O**” appears after the hex number in the filename, the file is played on a *opening* of the corresponding optional contact input module input or serial command. The sound will start when the associated contact input goes from a closed to an open state.

If a “**P**” after the hex number in the filename, the file will play only while the “**C**” or “**O**” state is true. If “**C**,” it will play while the associated contact input is closed and it will stop playing the sound when the input is opened. If “**O**,” it will play while the associated contact input is open and it will stop playing the sound when the input is closed.

If an “**R**” appears in the filename after the hex number, the sound is *repeated*, that is to say that the same sound file plays over and over. If used in conjunction with the “**P**” command, the sound file will repeat the entire time the “**C**” or “**O**” input condition is true. If “**C**,” the entire time the input is closed, if “**O**,” the entire time the input is open. If triggered without a “**P**” the file will play indefinitely or until another sound is played.

If an “**S**” is used in the filename, the on-board Push To Talk relay and a contact closure corresponding to the input number on an optional plug-in board will be activated when the file is played.

If a “**B**” is used in the filename, the sound is considered to be a *background* sound. Background sounds play when no other sound is playing, and multiple background sounds are played in the numerical sequence of the two digit input number, starting with the special file BACKGRND.WAV which is treated as if it were file 00.WAV. Sounds that are marked as Background may not be marked for contact closure or opens, play while or repeat. Such markings will be ignored.

If an “**N**” is used in the filename, the sound is considered to be a *non-interruptable* sound. Sounds that are marked as non-interruptable will play to completion before any other sounds are played. Contact events that occur while non-interruptable sounds are playing are discarded unless the SaveNIContacts entry in the optional CFSOUND.INI file [Contacts] section is set to TRUE. Sounds that are marked as non-interruptable may not be marked for Repeat or as a Background sound. Such markings will be ignored.

Sample Filenames

- “01C.WAV” This file would play once, when triggered by optional contact input #1 closing.
- “02O.WAV” This file would play once, when triggered by optional contact input #2 opening.
- “04CP.WAV” This file would play once, when triggered by optional contact input #4 closing and play only while input #4 was closed without repeating.
- “07CPR.WAV” This file would play and repeat, the entire time optional contact input #7 was closed.
- “01OPR.WAV” This file would play and repeat, the entire time optional contact input #1 was open.
- “08CNS.WAV” This file would play once, without being interrupted when triggered by optional contact input #8 closing. The on-board PTT relay would be activated for the duration of the sound as well as optional contact output #8.
- “7EB.WAV” This file would play as a background sound whenever any other sound was not playing.
- “7FB.WAV” This file would also play as a background sound after the preceding file 7EB.WAV had played.
- “19C.WAV” This file would play once, when triggered by contact input #25 on the PTT connector on the back of the CFSound-II closing.

Compact FLASH Card Loading and Scanning

Once programmed with the desired sounds and correct filenames, the card is inserted into the slot on the front of the **Compact FLASH II Digital Audio System**. The green LED indicator should begin to flash indicating that the unit is scanning and loading the files from the card. When the card is correctly loaded and scanned, the green LED should glow steadily. Any problems encountered while reading the CF card are indicated by a repeating sequence of flashes on the red LED indicator.

The card may be removed at any time. In that case, both the green and red LED indicators should turn off.

The optional contact inputs are ‘debounced’ by sampling them using a periodic interrupt. The sample rate is 50 Hertz (20 milliseconds). Valid input closures are detected by reading a 0 - 0 - 1 (open, open, closed) sample sequence. Valid input opens are detected by reading a 1 - 1 - 0 (closed, closed, open) sample sequence.

The status of the unit while loading and scanning is also displayed via messages sent out the serial port. Using a PC with “HyperTerminal” or other terminal emulator software connected to the **Compact FLASH II Digital Audio System** serial port the card loading and scanning messages may be viewed. The serial data format is 2400 baud, 8 data bits and 1 stop bit. Samples of these messages are shown in the Appendix.

Background Audio Tracks

If a sound file is copied onto the Compact FLASH Card with the name “BACKGRND.WAV,” this sound file will play only when no other audio tracks are being played. It will cease playing when the command is issued to play another sound file. Once that sound file is finished playing, the BACKGRND.WAV file will resume play from the point where it was interrupted and continue to play until again interrupted.

In addition, any files with the “B” token in their filename after the numeric portion will also be played in numerical sequence as background sounds, following the BACKGRND.WAV file.

CFSOUND.INI File

Support is provided to control various operating parameters by the inclusion of a CFSOUND.INI file on the CF card. When a card is inserted and after all sound files have been loaded, the CFSound II unit looks for a file with this name, and reads initialization parameter entries from it. Parameter entries are organized into sections with a section name in brackets followed by one or more PARAMETER=VALUE lines. The Section Names and Parameter entries are case sensitive. A section or parameter may be commented out by placing a '#' character as the first character on the line.

Here is a list of currently supported Sections and Parameter entries:

<u>[Section] / Parameter</u>	<u>Description</u>
[Comm]	Communications Port Section
BaudRate=dddddd	Sets the serial port baudrate to the decimal value ddddd. Default=2400.
[DEBUG]	Debug Section
ShowStartStop=TRUE/FALSE	Enables RS-232 message display of sound start/stop events. Default=FALSE.
[Background]	Background Section
BackgroundDelay=dddddd	Sets the delay in seconds between background sound playouts to the decimal value ddddd. Default=0.
[Quiz]	Quiz Section
QuizMode=TRUE/FALSE	Enables Quiz/Kiosk mode of operation. Default=FALSE.
QuestionContacts=dd	Sets the number of question contacts to the decimal value dd. Default=4.
AnswerContacts=dd	Sets the number of answer contacts to the decimal value dd. Default=4.
NoAnswerTimeout=dd	Sets the delay in seconds between the end of the question sound and the timeout answer sound to the decimal value dd. Default=5.
AwaitAnswerSound=xx	Sets the hexadecimal sound number xx to play after the question sound before the timeout answer sound. Default=0 (no sound).
AnswerWithoutQuestionSound=xx	Sets the hexadecimal sound number xx to play if an answer contact is activated before a question contact. Default=0 (no sound).
[Contacts]	Contacts Section
Force=TRUE/FALSE	Setting this value to TRUE restores the original CFSound contact behavior wherein the contact's active status is 'forced' upon reset, power-up or card-insertion. This will cause associated sound activation if the contact was active. Setting this value to FALSE (the default) causes the new behavior wherein the contact's current status is sampled upon reset, power-up or card-insertion. This will cause no associated sound activation until the contact is re-activated. Default=FALSE.
SequenceContactNumber=dd	Sets the number of the contact that will play sounds in sequence to the decimal value dd. Default=0 (no sequencing)
FirstSoundNumber=dd	Sets the first sound number that will be played in sequence to the decimal value dd. Default=1 (sound #1)

[Section] / Parameter	Description
LastSoundNumber=dd	Sets the last sound number that will be played in sequence to the decimal value dd. Default=127 (sound #127)
SaveNIContacts=TRUE/FALSE	Setting this value to TRUE will remember any contact events that occur while a non-interruptable sound is playing. Note that this can cause a non-interruptable sound to play again if its contact is re-activated while it is playing. Default=FALSE
OutputContactModulus=dd	Setting this value to non-zero will cause the output contacts associated with sounds to repeat on the modulo value if QuizMode=FALSE . Example: OutputContactModulus=4 activates contact outputs 1 through 4 for sounds 1 through 4, contact outputs 1 through 4 for sounds 5 through 8, etc. Default=0
RandomSequence=TRUE/FALSE	Setting this value to TRUE will cause each activation of the SequenceContactNumber to play a random sound from the range FirstSoundNumber to LastSoundNumber. Default=FALSE
OffsetContactNumber=dd	Sets the number of the contact that will offset the sounds associated with the other contacts by ContactOffsetAmount to the decimal value dd. Does not affect Sequence or Quiz mode. Default=0 (no offset)
ContactOffsetAmount=dd	Sets the value that will be added to the input contact number when the OffsetContactNumber input is active, to offset the actual sound number that will play to the decimal value dd. Does not affect Sequence or Quiz mode. Default=0 (no offset amount)
AutoplayEntireSequence=TRUE/FALSE	Setting AutoplayEntireSequence to TRUE causes the entire sequence of sounds to be played once whenever the SequenceContactNumber activates one time. Default=FALSE (no autoplay)
RepeatPlayWhileSound=TRUE/FALSE	Setting RepeatPlayWhileSound to FALSE causes sounds flagged as Play While ('P' attribute) to not repeat while the activating contact is active. Default=TRUE (repeat while active)

Quiz / Kiosk Mode

The CFSound II with version 2.1 firmware or higher, may be configured to operate in a Quiz or Kiosk mode. This mode is enabled by including the requisite [Quiz] section entries in a CFSOUND.INI file on the card with the **QuizMode=TRUE** parameter.

In Quiz mode, Question contacts trigger the Question sounds, then Answer contacts trigger the Answer sounds related to the question that was activated. The Question contacts start at contact 1 and are contiguous through the number of **QuestionContacts=dd** parameter. The Question sounds would be named 01C.WAV, 02C.WAV, ... , xxC.WAV where xx is the hexadecimal equivalent of the **QuestionContacts=dd** parameter.

The Answer contacts immediately follow the Question contacts and are contiguous through the number of **AnswerContacts=dd** parameter. Each Answer contact is associated with an array of Answer sounds, with the offset into the array controlled by the Question contact that was activated. There is also an additional timeout Answer sound that will be played if an Answer contact is not activated within the **NoAnswerTimeout=dd** seconds interval.

Due to the maximum number of sounds being limited to 146 including the optional BACKGRND.WAV sound the following limitation must be observed:

QuestionContacts x (AnswerContacts + 1) <= 145

The next table shows a sample of Quiz mode sound file names and their contact associations given the following values for the [Quiz] section parameters: **QuestionContacts=4**, **AnswerContacts=4**, **AnswerWithoutQuestionSound=7E** and **AwaitAnswerSound=7F** in the CFSOUND.INI file. Notice how the Answer contacts play different sounds depending upon what Question contact was activated:

Sound Filename	Contact #	Description
01CS.WAV	1	Question #1 sound
02CS.WAV	2	Question #2 sound
03CS.WAV	3	Question #3 sound
04CS.WAV	4	Question #4 sound
05CS.WAV	5	Question #1 Answer #1 sound
06CS.WAV	6	Question #1 Answer #2 sound
07CS.WAV	7	Question #1 Answer #3 sound
08CS.WAV	8	Question #1 Answer #4 sound
09.WAV	-	Question #1 no Answer Timeout sound
0ACS.WAV	5	Question #2 Answer #1 sound
0BCS.WAV	6	Question #2 Answer #2 sound
0CCS.WAV	7	Question #2 Answer #3 sound
0DCS.WAV	8	Question #2 Answer #4 sound
0E.WAV	-	Question #2 no Answer Timeout sound
0FCS.WAV	5	Question #3 Answer #1 sound
10CS.WAV	6	Question #3 Answer #2 sound
11CS.WAV	7	Question #3 Answer #3 sound
12CS.WAV	8	Question #3 Answer #4 sound
13.WAV	-	Question #3 no Answer Timeout sound
14CS.WAV	5	Question #4 Answer #1 sound
15CS.WAV	6	Question #4 Answer #2 sound
16CS.WAV	7	Question #4 Answer #3 sound
17CS.WAV	8	Question #4 Answer #4 sound
18.WAV	-	Question #4 no Answer Timeout sound
7EC.WAV	5,6,7,8	Answer Without Question Sound
7F.WAV	-	Awaiting Answer Sound

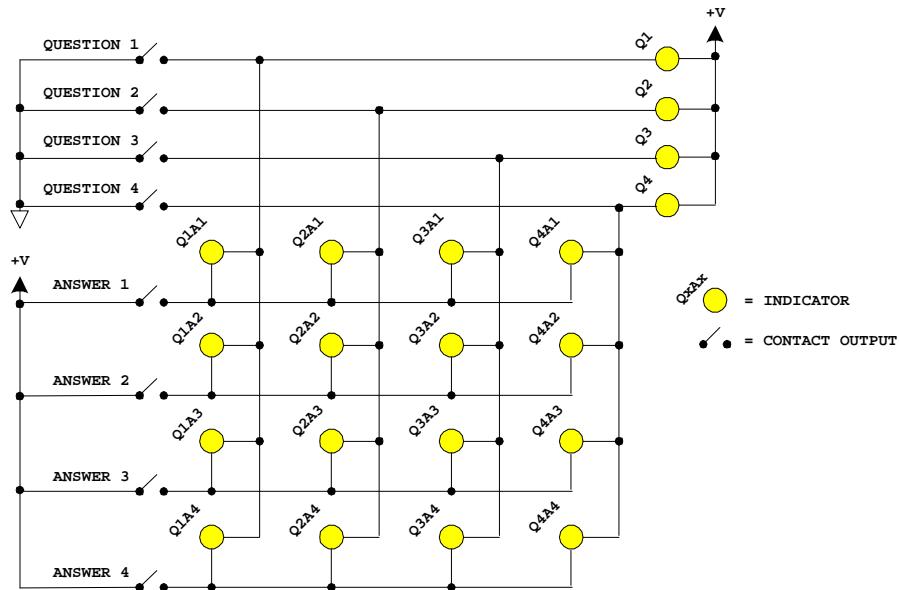
Note that while these sample files all have the 'C' contact closure token in the filename, the 'O' contact open token may be used as well. Sound files that are played without a direct contact closure do not require these tokens.

There are two additional, optional sounds that may be played in Quiz mode. The **AwaitAnswerSound=xx** parameter, if xx is non-zero, causes sound file xx.WAV to be played after the question sound while awaiting an Answer contact activation. In the preceding example, this would play sound file 7F.WAV. The duration of this optional sound or the **NoAnswerTimeout=dd** seconds parameter, whichever is greater, determines the interval before the associated no answer timeout sound file is played. The **AwaitAnswerSound** should not be marked as non-interruptable.

The **AnswerWithoutQuestionSound=xx** parameter if xx is non-zero causes sound file xx.WAV to be played if an Answer contact is activated before a Question contact to prompt user to select question first. In the preceding example, this would play sound file 7EC.WAV when any Answer contact is activated before a Question contact.

A Kiosk mode of operation may be obtained by setting the **NoAnswerTimeout=dd** parameter to zero. This effectively disables the **AwaitAnswerSound** and all of the **No Answer Timeout sounds** simply allowing the Question contact to select different sets of Answer contacts sounds such as various languages or features. After a Question contact is activated, it remains active selecting a given set of Answer contact sounds until another Question contact is activated.

Contact outputs may also be activated for Question and Answer sounds by using the 'S' token in the filename. In Quiz mode, Question contact outputs remain active for the duration of the Question/Answer cycle. In Kiosk mode, the last Question (answer selection) contact output remains active until another Question input is activated. Note that the Answer contact outputs associated with the Answer contact inputs are used for each set of answers, independent of which Question contact is activated. Unique Answer contact outputs may be achieved by constructing a matrix using the Question contact outputs as Rows and the repeating Answer contact outputs as Columns (or vice versa). This is shown in the following diagram:



Matrix Wiring of Question/Answer Contact Outputs

The **BackgroundDelay=dddddd** parameter can be used in conjunction with one or more background sounds to provide a delay between sounds, so that they can be used as an attention getting attract mode.

Sequential Sound Playout

The CFSound II with version 2.5 firmware or higher, may be configured to allow a single contact closure (or opening) to play several sounds in sequence; one each time it is closed (or opened). This mode is enabled by including the requisite [Contacts] section entries in a CFSOUND.INI file on the card with the **SequenceContactNumber=dd** parameter set to a non-zero, valid contact number from 1 to 8 (1 to 24 with Contact Sense 24 module).

The range of sounds that may be played in sequence may be constrained by the addition of two other parameter entries in the [Contacts] section. The **FirstSoundNumber=dd** parameter controls the starting sound number of the sequence, and the **LastSoundNumber=dd** parameter controls the ending sound number in the sequence. Setting the **FirstSoundNumber** parameter to a higher number than the **LastSoundNumber** will cause the sounds to be played in reverse sequence, higher number to lower number. The sequence wraps around to the beginning or end.

For example to play normal sounds on the first 7 contacts, and sequence through 16 different sounds for the 8th contact, a card would be prepared with the 23 sound files named 01C.WAV – 17C.WAV. Then a CFSOUND.INI file would be added to the card with the entries:

```
[Contacts]
SequenceContactNumber=8
FirstSoundNumber=8
LastSoundNumber=23
```

The first seven contacts would each play their associated sound. Each closure of contact 8 would play a sound from 08C.WAV – 17C.WAV then repeat from 08C.WAV – 17C.WAV in sequence. To reverse the order of the sound playout set the **FirstSoundNumber=23** and the **LastSoundNumber=8**.

To cause the entire sequence of sounds to be played once whenever the **SequenceContactNumber** is activated, set the **AutoplayEntireSequence** parameter to TRUE. For example a museum exhibit needs to play a sequence of 8 sounds when triggered with each sound activating a relay output while it is playing. A CF card would be prepared with the 8 sound files in the sequence named 01CNS.WAV – 08CNS.WAV. Then a CFSOUND.INI file would be added to the card with the entries:

```
[Contacts]
SequenceContactNumber=1
FirstSoundNumber=1
LastSoundNumber=8
AutoplayEntireSequence=TRUE
```

Whenever contact 1 is closed, the eight sounds 01CNS.WAV – 08CNS.WAV would play in sequence, each sound activating its output relay. This feature only works for non-random sequences and requires version 2.12 or higher firmware. The addition of the non-interruptable attribute to the file names prevents multiple contact 1 closures from prematurely advancing the sequence.

RS-232 Serial Port Protocol

Sound play-out may also be controlled via commands received via the serial port. The data format is 2400 baud, 8 data bits and 1 stop bit. Sounds may be started, queued and stopped using a simple ASCII protocol:

```
<SOH> p <+/-/&/~/!> XX <ETX>
```

where:

<SOH>	= ASCII Start of Header character 01_{16} (Ctrl-A)
p	= ASCII lower case letter 'p'
<+/->	= ASCII plus character '+' to start a sound, minus character '-' to stop a sound
<&/~/!>	= ASCII ampersand character '&' to queue a sound, tilde character '~' to flush the queued sounds, exclamation character '!' to stop the current playing sound and flush the queued sounds
XX	= ASCII two digit hexadecimal number XX_{16} of the sound (01, 02, . . . , 7E, 7F)
<ETX>	= ASCII End of Text character 03_{16} (Ctrl-C)

Up to 128 sounds may be queued. Sounds that are queued are played in succession in the order that they were queued only when no other sound is playing until the queue is exhausted or flushed. Note that this can occur at the end of the current background sound file before it repeats. A background or other sound that is playing may be stopped by issuing a "<SOH>p-00<ETX>" serial command sequence to allow queued sounds to be heard. The background sound will be resumed when the queue has emptied.

Card Requirements

The **Compact FLASH II Digital Audio System** was designed to use only Compact FLASH Cards (CF Cards) that support the True IDE ATA mode at 5 volt operation. Tested cards include San Disk, Kingston, Techworks, Dane-Elec, Memorex and Hitachi. The San Disk card was tested, since they manufacture 90% of the retail cards on the market. Compact FLASH Cards are available in many local electronics or camera stores or can be ordered directly from ACS.

LEXAR Media Compact Flash cards are not 100% compatible in True IDE mode and as such are NOT COMPATIBLE with the Compact Flash II Digital Audio System.

By default, Windows XP will format any Compact Flash card of 64MB or more with FAT32 format. The ACS Compact FLASH II Digital Audio System uses the FAT (FAT16) format and can not operate with a FAT32 formatted card. You must select FAT file system to format your Compact Flash card in a Windows XP PC, *not* FAT32.

Diagnostic LED Codes (Red & Green LEDs)

Green LED

- Flashes twice a second while scanning or loading CF Card files.
- Steady on when loaded, ready or playing.

Red LED Error Flashes

- Once every six seconds - problem w/ATA interface, non-compatible CF card, bad CF card.
- Twice every six seconds - error opening file for that input, incorrect filename, file does not exist.
- Three times every six seconds - error when reading file, file corrupted, unsupported file type.
- Four times every six seconds - no valid “*.WAV” files of a compatible format found on the CF Card.
- Five times every six seconds – no valid FAT file system found on the CF Card.
- Six times every six seconds – error seeking when resuming sound.

Power Requirements

The idle current of a CFSound II unit with an 8MB Kingston CF card installed is ~170mA @ 12VDC input.

With a background sound playing at a moderate level, the current consumption averages ~250mA @ 12VDC.

Digital Audio Recording

Initially make all recordings (either music or voice) with a sample rate of at least 22KHz and 16-bit. This high quality, first-generation recording will later provide the greatest bandwidth and produce the best sounding final audio.

Recommended Recording Procedure

1. Record any audio clip of your choosing @ 22 or 44KHz, 16-bit monaural or Stereo. Make sure that your samples do not clip (go above or below the sample window). The peaks of your audio clip, should, however, fill vertically at least 95% of the sample window. If they do not, you will need to boost your input signal by using a pre-amp or by some other means.
2. Edit the clip until satisfied. You can use a sound editor or filter program, such as Syntrillium's Cool Edit to normalize the amplitude of the sample. Use this command cautiously though, since it also tends to amplify noise levels. It is always preferable to re-record your sound clip at a higher level to achieve better fidelity.
3. After your editing is complete and you are ready to make a file for the **Compact FLASH II Digital Audio System**, use Cool Edit or a similar program to convert or down sample it to a 22KHz, 8-bit monaural or 11KHz 8-bit Stereo clip.

It is imperative that all initial editing and filtering be done to the audio clip while it is formatted at 16-bit and 22KHz or greater. Another important thing to remember when recording sound is that the background noise is amplified 256 times when down converting from 16 bit to 8 bit. It is, therefore, critical to the quality of your final sound that there is no background noise while you are recording your clips or samples. The quiet noise of the fan on a PC for example, when amplified 256 times, can produce a very noticeable hiss or static sound in the recorded clips or samples. Please be certain that your recording environment is absolutely quiet.

Also remember that digital play-out devices do not introduce any static of their own other than quantization noise. They only play back exactly what was recorded. If there is excessive static in the audio clip or sample, then there was probably static in it originally.

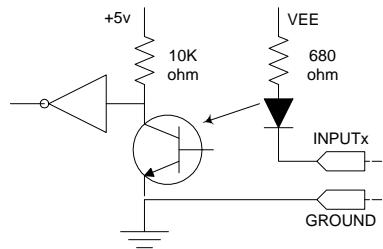
Optional Plug-in Modules

The **Compact FLASH II Digital Audio System** supports optional plug-in modules to provide additional Contact I/O capability.

CFSound II Contact Sense 8 Module

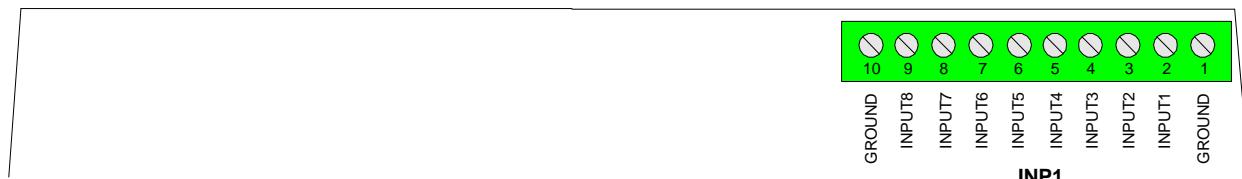
This module provides eight contact inputs. The contacts are numbered 1 through 8 and are associated with sound files named 01.WAV through 08.WAV.

The Cathodes of the LEDs in the optocouplers are connected to the Input connector port pins. The Anodes of the LEDs in the optocouplers are connected to an internal 12VDC power supply, with a 680 ohm current limiting resistor in series. The output transistor of each optocoupler has a 10K pullup resistor on its collector, with the emitter connected to ground and is buffered by an inverting gate. The following diagram is representative of one input:



An input is activated by sinking current from the corresponding input pin to ground. A Ground connection is supplied on pins 1 and 10 of each Input connector for this purpose. The input current sink requirement is approximately 15 mA.

The Input connector INP1 consists of a 10 pin, 2 piece terminal block style pluggable connector. The pinout is shown in the following diagram and table:

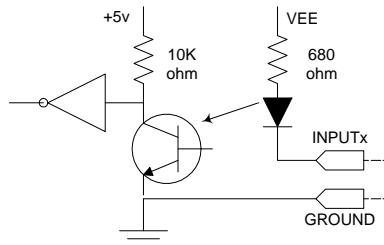


INP1 Pin #	Signal
1	GROUND
2	INPUT1
3	INPUT2
4	INPUT3
5	INPUT4
6	INPUT5
7	INPUT6
8	INPUT7
9	INPUT8
10	GROUND

CFSound II Contact I/O 8 Module

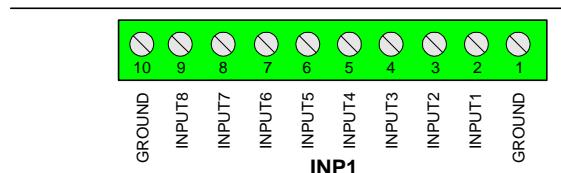
This module provides eight contact inputs and eight contact outputs. The contacts are numbered 1 through 8 and are associated with sound files named 01.WAV through 08.WAV. If an 'S' token is present in the sound filename, the associated output contact and on-board PTT relay will be activated whenever the sound is played.

The CFSound II Contact I/O 8 Module provides eight opto-isolated current inputs. The Cathodes of the LEDs in the optocouplers are connected to the Input connector port pins. The Anodes of the LEDs in the optocouplers are connected to an internal 12VDC power supply, with a 680 ohm current limiting resistor in series. The output transistor of each optocoupler has a 10K pullup resistor on its collector, with the emitter connected to ground and is buffered by an inverting gate. The following diagram is representative of one input:



An input is activated by sinking current from the corresponding input pin to ground. A Ground connection is supplied on pins 1, and 10 of the Inputs connector INP1 for this purpose. The input current sink requirement is approximately 15 mA.

The Inputs connector INP1 is a 10 pin, 2 piece terminal block pluggable style. The pinout is shown in the following diagram and table:

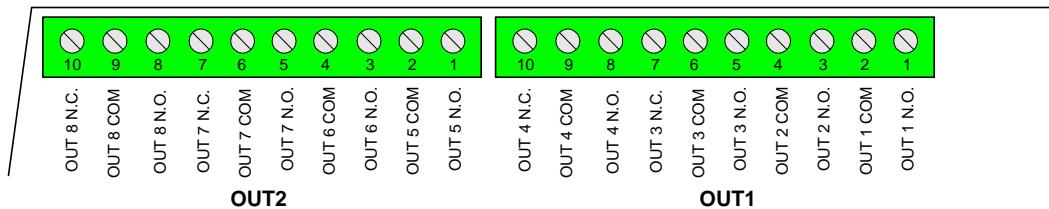


INP1 Pin #	Signal
1	GROUN D
2	INPUT1
3	INPUT2
4	INPUT3
5	INPUT4
6	INPUT5
7	INPUT6
8	INPUT7
9	INPUT8
10	GROUN D

The CFSound II Contact I/O 8 Module provides eight normally open (N.O.) relay contact outputs. Outputs 3, 4, 7 and 8 also provide the normally closed (N.C.) relay contact outputs.

Each contact is rated at 1A @ 30VDC.

The Output connectors OUT1 and OUT2 consist of two 10 pin, 2 piece terminal block style pluggable connectors that are mounted adjacently. The pinout is shown in the following diagram and table:

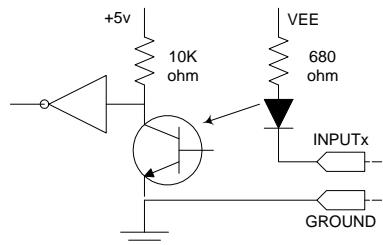


OUT2 Pin #	Signal	OUT1 Pin #	Signal
1	OUT 5 N.O.	1	OUT 1 N.O.
2	OUT 5 COM	2	OUT 1 COM
3	OUT 6 N.O.	3	OUT 2 N.O.
4	OUT 6 COM	4	OUT 2 COM
5	OUT 7 N.O.	5	OUT 3 N.O.
6	OUT 7 COM	6	OUT 3 COM
7	OUT 7 N.C.	7	OUT 3 N.C.
8	OUT 8 N.O.	8	OUT 4 N.O.
9	OUT 8 COM	9	OUT 4 COM
10	OUT 8 N.C.	10	OUT 4 N.C.

CFSound II Contact Sense 24 Module

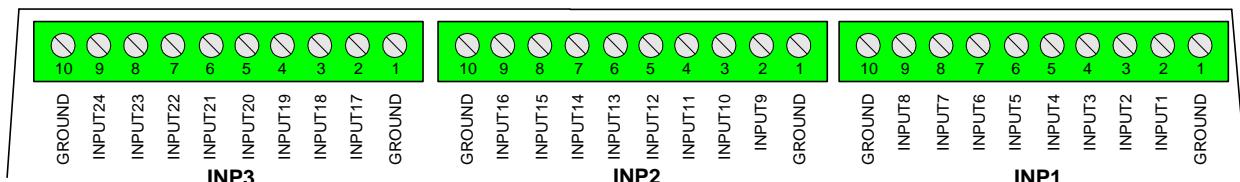
This module provides twenty four contact inputs. The contacts are numbered 1 through 24 and are associated with sound files named 01.WAV through 18.WAV.

The Cathodes of the LEDs in the optocouplers are connected to the Input connector port pins. The Anodes of the LEDs in the optocouplers are connected to an internal 12VDC power supply, with a 680 ohm current limiting resistor in series. The output transistor of each optocoupler has a 10K pullup resistor on its collector, with the emitter connected to ground and is buffered by an inverting gate. The following diagram is representative of one input:



An input is activated by sinking current from the corresponding input pin to ground. A Ground connection is supplied on pins 1 and 10 of each Input connector for this purpose. The input current sink requirement is approximately 15 mA.

The Input connectors INP1, INP2 and INP3 consist of three 10 pin, 2 piece terminal block style pluggable connectors mounted adjacently. The pinout is shown in the following diagram and table:



INP3 Pin #	Signal	INP2 Pin #	Signal	INP1 Pin #	Signal
1	GROUND	1	GROUND	1	GROUND
2	INPUT17	2	INPUT9	2	INPUT1
3	INPUT18	3	INPUT10	3	INPUT2
4	INPUT19	4	INPUT11	4	INPUT3
5	INPUT20	5	INPUT12	5	INPUT4
6	INPUT21	6	INPUT13	6	INPUT5
7	INPUT22	7	INPUT14	7	INPUT6
8	INPUT23	8	INPUT15	8	INPUT7
9	INPUT24	9	INPUT16	9	INPUT8
10	GROUND	10	GROUND	10	GROUND

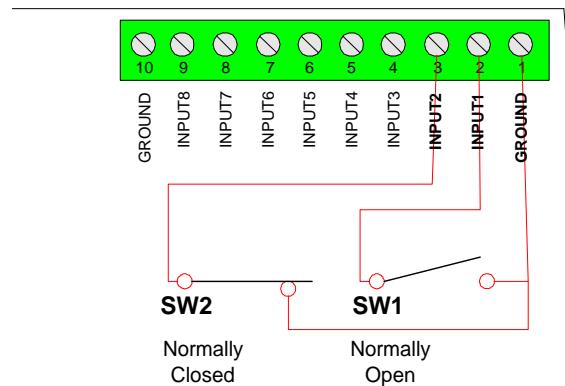
Contact Wiring

On the optional Contact Modules, switch contacts are wired between the Input pin number and a ground located on either end of the connector.

Switches may be one of two forms: Normally Open (N.O.) or Normally Closed (N.C.). Switches that are Normally Open have no electrical connection between the switch terminals unless the switch is activated closed. Switches that are Normally Closed have an electrical connection between the switch terminals unless the switch is activated open. Infrared motion detectors designed for security systems are normally closed so that cut wires may be detected.

The CFSound II Contact Modules can work with either form of switch by using the appropriate filename letter token in the filename. A sound can be played when a Normally Open switch *Closes* by placing the ‘C’ letter token after the sound number in the sound filename. In a similar fashion, a sound can be played when a Normally Closed switch *Opens* by placing the ‘O’ letter token after the sound number in the sound filename.

In the following diagram, Input1 is wired to a Normally Open switch SW1, and Input2 is wired to a Normally Closed switch SW2. The sound file to be played when SW1 closes would be named 01C.WAV. The sound file to be played when SW2 opens would be named 02O.WAV.



Frequently Asked Questions

Q. The RED LED flashes four times at power up?

Q. The RED LED flashes two times when I try to play a sound, and no sound plays? Why is my CFSOUND.INI file not making the CFSound II perform using the features I have specified ?:

This is by far our most commonly asked question. 99% of time, it is because your files have double extensions. I.E. "01C.WAV.WAV" or "CFSOUND.INI.INI". Later versions of Windows hide the file extensions in explorer, so by putting ".WAV" or ".INI" in the filename, you are actually causing a duplicate file extension.

To turn the hiding of file extensions off in Windows XP, do the following:

Click on Start

Then Click on Control Panel

Next, Click on Folder Options

Click on the View tab on the Folder Options Window

Scroll down to "Hide Extensions for Known File Types"

Click on the Check box to the left of it to remove the check mark

Click on the Apply Button

Now the complete file names will be displayed in Explorer, and any folder, and you can remove the duplicate file extensions

Q. What Inputs and outputs are available on the main CFSound board, without adding I/O Modules ?:

The CFSound main board has a single PTT(Push To Talk) Relay for an output and a single contact input that appears as Contact #25 for an input. Additional Inputs and outputs are added by adding I/O Modules. The CFSound II can accept one I/O module, along with one amplifier.

Q. I have a motion sensor tied to an old industrial looping tape deck. The first time the sensor is broken it will play the first of three tracks, the second time the same sensor is broken it plays the second track on the tape and the third time it plays the third track. Can your machine duplicate this task ?:

Yes, the CFSound II can function as you described. You make three entries into the [CONTACTS] section of the CFSOUND.INI file that you place on the Compact Flash Card.

[Contacts]

SequenceContactNumber=dd

FirstSoundNumber=ff

LastSoundNumber=ll

--> Setting the SequenceContactNumber to anything greater than zero, causes the input # dd to operate in a sequencing mode.

--> FirstSoundNumber sets the first wav file number to be played in the sequence as # ff.

--> LastSoundNumber sets the last wav file number to be played in the sequence as # ll.

You would replace dd with the contact input you wanted to function this way, ff with the first sound that you wanted it to play, and ll with the last sound you want it to play. Every time there is a contact closure on the input specified in dd, it will play the files from ff to ll in sequence, then start over again at ff, and keep repeating.

Q. Can I use a momentary contact closure to initiate a message, but the message needs to stop by using another momentary closure (or open?). So, a momentary to initiate, and a momentary to stop any message that is in progress ?:

Yes, the CFSound II can function as you described. A triggered message that is playing could be 'stopped' by activating another short silent message associated with another contact input.

Q. Can the Input ports be momentary, or do they need to remain in a closed or open ?:

It depends upon what you're trying to do. If you want a message that is triggered by a contact closure (or opening) you only need a momentary switch.

If you want the sound to only play while the input is closed (or open) then you would need a latching switch.

Sounds are associated with inputs by the filenames of the sounds on the CF card. How they are activated is also controlled by 'decorating' the filenames with additional characters.

For example:

To play a sound whenever input 1 momentary closes, the sound file would be named 01C.WAV.

To play a sound whenever input 2 momentary opens the sound file would be named 02O.WAV.

To play a sound only while input 1 is closed, the sound file would be named 01CP.WAV.

To play a sound only while input 1 is open, the sound file would be named 01OP.WAV.

To repeat a sound only while input 1 is closed, the sound file would be named 01CPR.WAV.

Sounds are normally interruptable whenever another input is activated, but you can also add the character "N" to the filename and make the sound(s) non-interruptable.

You can also control the unit through the RS-232 port to start and stop sounds.

Q. Can a background sound or music be played until another input is triggered ?:

You can have one or more background sounds that play (with an optional delay in between). The background sound is interrupted by the contact input activated sound(s), and resumes (if active) when the contact sound is finished playing.

With a 512Meg CF card, you can have up to 384 minutes of hi-fi stereo audio - over six hours.

Q. Can we use basic switches to control our various functions. There are no logic circuits, or anything "hi-tech" in this design ?:

You shouldn't need any external logic circuits. We use 'dry' contact closures for the inputs and provide 'dry' contacts for the output(s). Simple off the shelf switches will work just fine. Water tight / weather proof switches should be used if exposed to elements of weather.

Q. Can individual audio files/channels play simultaneously? i.e. can a file be triggered to play, overlapping a file or files that are already playing ?:

The CFSound cannot play files simultaneously, but it does offer several features that get around this limitation.

1) When the "BACKGRND.WAV" sound/file is playing, and it is interrupted by a contact input, it pauses the "BACKGRND.WAV" sound, plays the sound triggered by the contact input, then continues playing the "BACKGRND.WAV" sound from where it was interrupted.

2) When sounds are played by an external contact input/trigger, they can be configured by:

Characters in there filename to:

- Not interruptible
- Interruptible

Settings in the CFSOUND.INI file to:

- Store inputs/triggers, and play them when the current sound in finished
- Not store inputs/triggers while a sound is being played

Q. I am having addressing problems. With the 'old' units I was able to run a lead from common to '0nCPR' and have the unit start playing a track "n" automatically on power up. This does NOT happen with these units. It seems there is something wrong with addressing operation. Is there any adjustment I can make ?:

We did indeed make a software change that pre-reads the contact inputs on power-up to prevent automatic sound activation if contacts were already closed. We had a customer complain about this 'feature'.

To support this un-documented feature, we added an additional CFSOUND.INI file entry of:

[Contacts]

Force=TRUE/FALSE

--> Setting this value to TRUE restores the original behavior wherein the contact's active status is 'forced' upon reset, power-up or card-insertion. This will cause associated sound activation if the contact was active.

--> Setting this value to FALSE (the default) causes the new behavior wherein the contact's current status is sampled upon reset, power-up or card-insertion. This will cause no associated sound activation until the contact is re-activated.

Q. Do you have more detailed spec's on your amplifier module ?:

The amplifier is based upon a pair of National Semiconductor LM1875s and is roughly 7 watts (3.5W/channel) with a 12VDC power supply, or, roughly 15 watts (7.5W/channel) if a 24VDC supply is connected. Mono sounds play through both channels.

Q. How do I use the Windows Sound Recorder to make recordings ?:

1. Click on the Start Button, then Programs, then Accessories, then Entertainment, then Sound Recorder.
2. Click on Files, then Properties. A new window opens named "Properties For Sound".
3. Change the "Format Conversions" "Choose From" drop down box to "All Formats" or "Recording Formats".
4. Next, click on "Convert Now", a new window named "Sound Selection" opens.
5. Make sure "Format" is PCM.
6. Make Sure "Attributes" is set to one of the following.
 "22,050 Hz, 8 Bit, Mono 22KB/s"
 "16,000 Hz, 8 Bit, Mono 16KB/s"
 "11,025 Hz, 8 Bit, Stereo 22KB/s"
 "8,000 Hz, 8 Bit, Mono 8KB/s"
7. Click on the OK Button to close the "Sound Selection" window.
8. Click on the OK Button to close the "Properties For Sound" window.
9. To start recording, click on the button with the red dot(record).
10. To stop recording, click on the button with the black square(Stop).
11. Once you have finished recording, Click on File, then Save As, and name your file how you want it to play.
12. I.E. "BACKGRND.WAV" to play when there is no switch closures.
13. I.E. "01C.WAV" to play when input #1 closes.
14. Copy the file to the Compact Flash Card.

You're done.

NOTE: If you click on File, then click new, you will have to set the File properties again before recording.

Q. Can a output contact be used to operate a lamp to indicate that the message has been triggered ?:

If you need individual lights that illuminate when the associated input is active, you would need the Contact I/O 8 module in the CFSound II. If you only need a single light that is illuminated whenever any sound is playing then you can use the PTT relay that is installed on the main unit and use a Contact Sense 8 or Contact Sense 24 module.

Q. Assuming that an additional card is required for Inputs and/or Outputs - what if we require both an I/O card and the stereo amplifier in an enclosed unit ?:

The CFSound II can accept one I/O module, along with one amplifier.

Q. When I try to use a Lexar CF Card, the CFSound II Flashes the RED Error light once every six seconds ?:

LEXAR Media Compact Flash cards are not 100% compatible in True IDE mode and as such are NOT COMPATIBLE with the Compact Flash II Digital Audio System.

We have asked Lexar for technical information on this incompatibility, and they have not responded.

Q. We have a client who has a specific need -- an interval timer function. They want their very short message to go off every 30 minutes (not seconds). Does your equipment have a provision for an interval timer ?:

By placing a file named CFSOUND.INI on the compact flash card along with your sound files, you can control several operating features of the unit.

For your application the following entry would provide the 30 minute delay between the background sound:

[Background]

BackgroundDelay=1800

This .INI file option sets the delay in seconds between background sound playouts to the decimal value of 1800 seconds (30 minutes). The default is 0 seconds (no delay). This is not a high-resolution timer and may be affected by other sounds playing while the timer is active.

Q. I have saved my file as 07CPR.wav and hooked up an alternating switch to Terminal one on INP1 and the other side of the switch to the eighth terminal on INP1 (page 17 of the manual) and I can start the audio by pushing the switch (closed), but if I push the switch again (open) it does not stop the audio. Why?? I thought maybe I had the wrong pin, but I tried it with different pins. What am I doing wrong ?:

We discovered a bug that developed when we added the quiz mode feature, in the CFSound II firmware, that made the Play while flag operate continuously. The sound would start, but then never stop after the contact was opened.

The firmware is upgradeable - the latest version with the fix is v2.30. The unit displays the version on power-up via the serial port to an attached PC using Hyperterminal.

Please visit <http://www.cfsound.com> and check the Downloads section for the latest version as well as the programming utility that will upgrade your unit. There is also a .PDF of the firmware revisions.

Q. We need to have a light associated with an individual "message initiate" momentary switch when that message is playing. Do we need only the Contact I/O 8 module, or do we also need the Contact Sense 8 ?:

The Contact I/O 8 has 8 contact sense inputs and 8 contact outputs. The Contact Sense 8 has 8 contact sense inputs and the Contact Sense 24 has 24 contact sense inputs.

You can only have a single contact module installed.

Q. I ordered an enclosed CFSoundII, and no volume control knob was installed/received ?:

We do not ship a volume knob with the CFSound II, since we feel that it is a set and forget type of application, and will not be adjusted once set.

Q. When wiring the power supply connector to the supplied power supply, which side is plus and which side is ground ?:

It is connected to a bridge rectifier internally, so it can be wired either way.

Q. Will it finish playing the current file even if another input is tripped half way through the file ?:

If you add an "N" in the filenames of your files, it marks the files as non-interruptible. Then setting SaveNIContacts= TRUE in the [CONTACTS] section of the ini file, will cause the contact closure during play out of the Non-Interruptible file to be saved until it completes playing.

Q. When I format my CF card under Windows XP, it uses FAT32, and does not work in the CFSound II ?:

By default, Windows XP will format any CompactFlash card of 64MB or more with FAT32 format. Digital cameras and the CFSound II use the FAT (FAT16) format and cannot operate with a FAT32 formatted card. Either format your CompactFlash card in a camera or select FAT file system (not FAT32) to format your CompactFlash card correctly in a Windows XP PC.

Q. How long does it take the CFSound-II to power-up and be ready to play sounds ?:

The unit must either scan the card upon power-up, or scan the card everytime a sound is initiated. We opted to scan the card upon power-up to enable each sound to be played quicker. Many of the delays upon power-up are due to having to support the variety of Compact Flash cards on the market - some of which require several seconds upon power-up before they are accessible. The firmware is designed to accommodate the worst case card, and we've had to slow it down in production as people encountered problems with various cards that couldn't be read. There is no firmware modification available that will shorten this delay. A freshly formatted small card with a limited number of sounds will scan the fastest. We have measured this delay at 20-30 seconds.

Q. How long does it take the CFSound-II to play a sound in response to a contact closure ?:

The unit debounces the contact inputs every 20mSEC. It requires a three read sequence of open, open, closed to qualify a contact input as valid. Then the file must be opened and buffering started. We have measured this delay at 160mSEC.

Sample Messages

The following sample messages were captured from the CFSound II serial port using Hyperterminal. The connection serial format is 2400 baud, 8 data bits, 1 stop bit.

Sample Power Up Message

```
Compact Flash II Sound v2.0
Copyright(c)1997-2002 by ACS, Sarasota, FL
ALL RIGHTS RESERVED
```

Sample Card Insertion Messages

SanDisk 8meg

```
CF Card Detected...
CF Card Powered...
CF Card Reset...
CF Card Reset Released...
SST delay
Found 1 devices on this ATA interface:
    Device 0 type is ATA.

Device 0 is the selected device...
Soft Reset...
ATA Set Features 8-bit...
ATA Identify Device...
Model Number:SanDisk SDCF8-8
Firmware Revision:vdd 1.00
Serial Number:          09305641112
# of Cylinders: 245
```

```
# of Heads: 2
# of Sectors/Track:32
# of Sectors:15680
Partition at Sector:32
```

Dane-Elec 16meg

```
CF Card Detected...
CF Card Powered...
CF Card Reset...
CF Card Reset Released...
SST delay
Found 1 devices on this ATA interface:
Device 0 type is ATA.

Device 0 is the selected device...
Soft Reset...
ATA Set Features 8-bit...
ATA Identify Device...
Model Number:TOSHIBA THNCF016MAA
Firmware Revision:1.1
Serial Number:STCB10M6001YE61850A0
# of Cylinders: 248
# of Heads: 4
# of Sectors/Track:32
# of Sectors:31744
Partition at Sector:32
```

Kingston 8meg

```
CF Card Detected...
CF Card Powered...
CF Card Reset...
CF Card Reset Released...
SST delay
Found 1 devices on this ATA interface:
Device 0 type is ATA.

Device 0 is the selected device...
Soft Reset...
ATA Set Features 8-bit...
ATA Identify Device...
Model Number:8MB CompactFlash Memory Card
Firmware Revision:CF-2A
Serial Number:SST21200000000002805
# of Cylinders: 245
# of Heads: 2
# of Sectors/Track:32
# of Sectors:15680
Partition at Sector:1
```

Memorex 8meg

```
CF Card Detected...
CF Card Powered...
CF Card Reset...
CF Card Reset Released...
SST delay
Found 1 devices on this ATA interface:
Device 0 type is ATA.

Device 0 is the selected device...
Soft Reset...
ATA Set Features 8-bit...
ATA Identify Device...
```

```

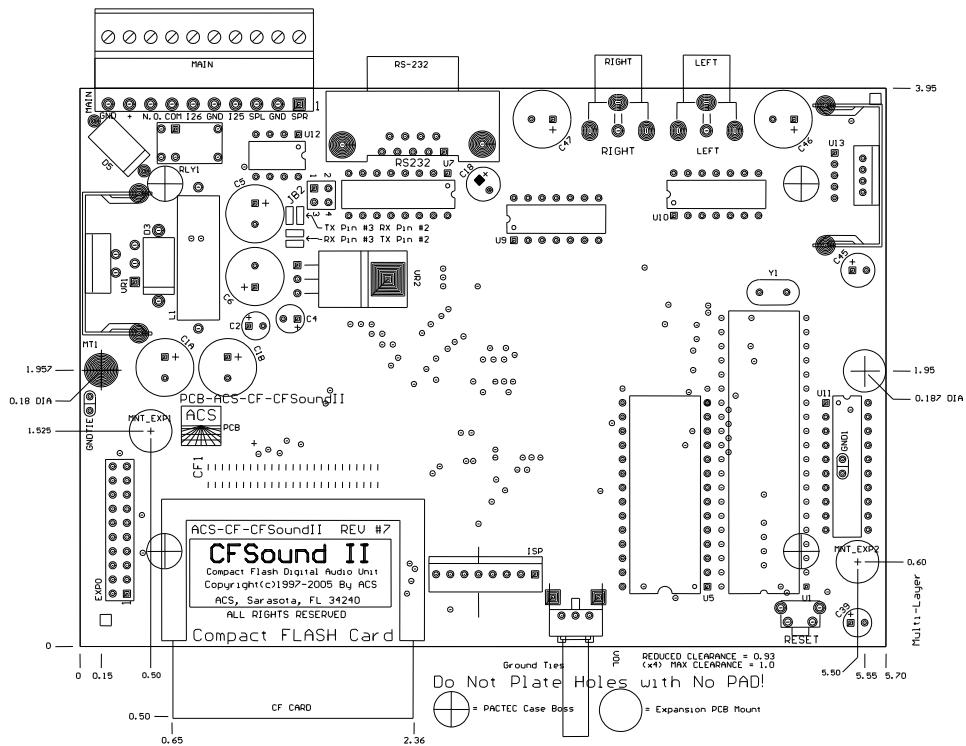
Model Number:CFA
Firmware Revision:4.0
Serial Number: CONED04B7311008C
# of Cylinders: 500
# of Heads: 4
# of Sectors/Track:32
# of Sectors:64000
Partition at Sector:32

```

Sample Directory Listing

#	File	Rate	Ofst	Length	C	O	P	R	S	B
0	BACKGRND.WAV	11025 Hz	44	1933416						B
1	01CS.WAV	16000 Hz	44	9238	C					S
2	02CS.WAV	16000 Hz	44	7270	C					S
3	03CS.WAV	16000 Hz	44	7186	C					S
4	04CS.WAV	16000 Hz	44	6380	C					S
5	05CS.WAV	16000 Hz	44	8726	C					S
6	06CS.WAV	16000 Hz	44	9500	C					S
7	07CS.WAV	16000 Hz	44	10950	C					S
8	08CS.WAV	16000 Hz	44	6100	C					S
9	09CS.WAV	16000 Hz	44	20998	C					S
10	0ACS.WAV	16000 Hz	44	8936	C					S
11	0BCS.WAV	16000 Hz	44	11340	C					S
12	0CCS.WAV	16000 Hz	44	9078	C					S
13	0DCS.WAV	16000 Hz	44	11864	C					S
14	0ECS.WAV	16000 Hz	44	12922	C					S
15	0FCSS.WAV	16000 Hz	44	10966	C					S
16	10CS.WAV	16000 Hz	44	13354	C					S
17	11CS.WAV	16000 Hz	44	14612	C					S
18	12CS.WAV	16000 Hz	44	12046	C					S
19	13CS.WAV	16000 Hz	44	13240	C					S
20	14CS.WAV	16000 Hz	44	7224	C					S
21	15CS.WAV	16000 Hz	44	12308	C					S
22	16CS.WAV	16000 Hz	44	12342	C					S
23	17CS.WAV	16000 Hz	44	12716	C					S
24	18CS.WAV	16000 Hz	44	12754	C					S
25	19CS.WAV	16000 Hz	44	13298	C					S
26	1ACS.WAV	16000 Hz	44	14434	C					S
27	1BCS.WAV	16000 Hz	44	14544	C					S
28	1CCS.WAV	16000 Hz	44	12686	C					S
29	1DCS.WAV	16000 Hz	44	14956	C					S
30	1ECS.WAV	16000 Hz	44	8064	C					S
31	1FCS.WAV	16000 Hz	44	11642	C					S
32	20CS.WAV	16000 Hz	44	13668	C					S

Mechanical



Firmware Revisions

Version	Date	Notes
1.0	3/16/98	First release for initial production CFSound.
1.1	4/20/98	Corrected filename parsing to utilize 2 digit ASCII hex, made check for COPR flags to start at filename[2].
1.2	4/24/98	Changed sound filename parsing to check for BACKGRND.WAV before looking for HEX filenames since BA is valid hex but out of MAX_SOUNDS range.
1.3	5/11/98	Corrected to allow hot swapping of CFA cards. You can now insert and remove cards while the unit is running and even playing a sound.
1.4	7/13/98	Corrected rdcf_error_message() to re-enable serial interrupts after displaying error messages. Modified StartSound() to clear the RED error LED display when a sound is successfully started.
1.5	5/28/99	Added a 'S' to COPR flags to activate a relay before playing a sound.
1.6	11/1/99	Increase MAX_SOUNDS from 48 to 65.
1.7	11/3/99	Removed the file info display in ShowDirectory() at card insertion to speed-up the power-up sequence.
1.8	3/8/00	Correct sound length to remove any zero pad bytes which can cause pops. Corrected sound size in StartSound() which incorrectly handled a sound size that ended in the second buffer which caused the sound to play off of the end and play out the LIST chunk at the end of the file as a short noise burst.
1.9	8/25/00	Increased command timeout 4x for slow SST controller cards.
1.10	1/23/01	Added delay macros in ataddrvr.c for commented out DELAY400NS calls. Disabled checking for ATA device 1 in ATA_Config() as it was screwing up CF cards using the Toshiba controller. Cleared BackgroundBufferNumber and BackgroundBufferNdx when card is removed so that new card's background will start from the beginning.
1.11	1/29/01	Added additional operators '&', '!' and '~' to the serial Play command. '&' appends the sound # to queue that is played out when the CFSound is idle. '~' clears the queue, and '!' stops the current sound and clears the queue. Up to 128 sounds may be queued. Increased MAX_SOUNDS from 65 to 129.
1.12	2/12/01	Added support for CFSOUND.INI configuration file read upon card insertion. Provides option to change baud rate.
1.13	4/4/01	Added improved serial I/O handling. Added DEBUG_SOUND_START_STOP manual compilation option below. Increased COMM_MAX_MSGSIZE to 52.
1.14	4/4/01	Modified a few instances of global interrupt disables to timer0 interrupt disable/enable. Changed some BOOLEANS to bit.
1.15	4/10/01	Changed all calls to ShowErrorAndTrace() and rdcf_error_message() to pass an int so that correct error messages are displayed. Changed all '\r' to '\r\n' sequences. Removed queue.* serial I/O handling since 8051 can't handle structures efficiently, replaced with macros in serial.h that expand in-line in serial.c. Replaced comm message handling with comm.* from vertical cpu codebase. Doubled size of serial receive buffer.
2.0	12/26/01	First version for new CFSound II and new I/O boards.
2.1	3/12/02	Added FAT12/16 detection and partition table display. Revised firmware to allow playing FAT16 formatted CF Cards, increasing sound playout capability from 32Mbyte to 512MByte. Added optional [DEBUG] section to CFSOUND.INI. ShowStartStop=TRUE causes serial i/o message to be displayed at current baud rate whenever a sound is started/stopped by whatever method. Removed DEBUG_SOUND_START_STOP manual compilation option. Added Quiz mode with separate [Quiz] CFSOUND.INI options. Added [Background] CFSOUND.INI options.
2.2	4/26/02	Increased power-up, reset and post-reset delays from 0.75 sec to 1.0 second for 32MB CTS controller in very slow Verbatim CF cards.
2.3	4/29/02	Added [Contacts] section to CONFIG.INI file with Force=TRUE option to cause contacts that are already active to start their associated sounds when the unit is reset. Corrected problem with PLAY WHILE sound flag continuously repeating once started.
2.4	6/18/02	Increased timeout in ATA_fast_pio_data_in_lba() to handle slower cards with larger directory structures.
2.5	7/1/02	Rewrote Quiz logic. Added QuestionTimeout entry to CFSOUND.INI file. Added sequential sound playout options to [Contacts] section of CFSOUND.INI file.
2.6	7/15/02	Corrections to background sound resume logic to fix channel

Version	Date	Notes
		swapping. Defaulted sequential sound playout option to default as disabled upon card insertion in case card is swapped after being enabled. Increased stack size.
2.7	7/25/02	Added an 'N' to COPRSB flags to mark a sound as non-interruptable. Non-interruptable sounds may not be marked for Repeat, and background sounds may not be marked as non-interruptable. Contact events that occur while non-interruptable sounds are playing are discarded unless the CFSOUND.INI file [Contacts] section entry SaveNIContacts=TRUE (default=False).
2.8	8/9/02	Added OutputContactModulus option to the [Contacts] section of the CFSOUND.INI file. Setting this to non-zero causes the output contacts to 'repeat' on the modulo value for sounds with the COPRSB 'S' flag set. With OutputContactModulus=4, sounds 1 through 4 activate output contacts 1 through 4, sound 5 through 8 also activate output contacts 1 through 4, etc..
2.9	10/22/02	Added RandomSequence option to the [Contacts] section of the CFSOUND.INI file. Setting this TRUE causes the sequential sound playout options to choose random sounds within the range of FirstSoundNumber and LastSoundNumber. Increased the maximum number of sounds from 128 to 145.
2.10	2/28/03	Added OffsetContactNumber and ContactOffsetAmount options to the [Contacts] section of the CFSOUND.INI file. Setting OffsetContactNumber to non-zero causes the input contacts to be offset by ContactOffsetAmount for playing the associated sound whenever the input contact at OffsetContactNumber is active.
2.11	6/17/03	Increased post CF Reset delay from 1 second to 2 seconds to allow use of extremely slow RiData CF cards. Added logic to try and skip over unnecessary RIFF chunks when parsing WAVE files.
2.12	8/12/03	Added AutoplayEntireSequence option to the [Contacts] section of the CFSOUND.INI file. Setting AutoplayEntireSequence to TRUE causes the entire sequence of sounds to be played once whenever the SequenceContactNumber activates one time.
2.13	9/30/03	Major changes to support new Servo I/O card. Removed free space display on powerup/card insertion that was taking a lot of time.
2.14	11/17/03	Changes to Servo I/O interrupt timing.
2.15	3/09/04	Added 6uSEC delay to the end of ATA_fast_pio_data_in() before the final status check to correct a problem reading SLOW FujiFilm CF cards.
2.16	6/14/04	Added support for Contact #25 on rear of CFSound II Revision 6.
2.17	7/28/04	Removed servo processing which was causing periodic sound lockups and looping. Restored sound buffer size to 11K each.
2.18	10/25/04	Fixed bug when checking Contact #25 for play while.
2.19	11/03/04	TestRawContact() for 25 th contact was checking debounced input instead of raw contact causing problem with Play While Closed functionality.
2.20	8/25/05	Added debouncing and processing of additional contact #26 for revision 7 boards.
2.21	12/19/05	Made active partition check default to partition 0 if no active partition can be found. Windows XP tends to not mark a partition as active on removable devices.
2.22	4/26/06	Corrected sound end if/else chain to skip checking for background sound start/restart when autoplay sequence is running. Also, restart and BackgroundDelayTimer when an autoplay sequence ends.
2.23	5/16/06	Corrected Play While sounds that are part of a sequence continuously trying to restart. Added RepeatPlayWhileSound (default=TRUE) Boolean option to the [Contacts] section of the CFSOUND.INI file to allow long play while sounds to be stopped without repeating.