HONORABLE JOHN P. ERLICK

IN THE SUPERIOR COURT OF THE STATE OF WASHINGTON
FOR KING COUNTY

ROBERT M. ARNOLD, derivatively on behalf of MOD SYSTEMS, INC. and BANANA CORPORATION, Plaintiffs, v. MARK PHILLIPS, KENNETH GORDON, JANDE DOE GORDON, and the marital community composed thereof; ANTHONY BAY, JANE DOE BAY, and the marital community composed thereof; MOD SYSTEMS, INC. and BANANA CORPORATION, Defendants.	NO. 09-2-07963-3 SEA DECLARATION OF MARK PHILLIPS

MARK PHILLIPS declares under penalty of perjury of the laws of the States of Washington as follows:

1. My name is Mark Phillips and I am over the age of 18, I have personal knowledge of the facts stated in this declaration and I am competent to testify to them.

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

2. I am a defendant in this case. I request that the MOD voting trust which is controlled by Kyleen Cane be terminated, that Ms. Cane and Mr. Bay be removed from the Demand Review Committee and that I be put back in control of the company I founded, of which I am the majority shareholder and for which I created the technology so that I can defend the claims. Toshiba and NCR would not have invested $35,000,000.00 in MOD but for me. If I am not there to protect MOD it will be bled dry. I will appoint an independent Demand Review Committee to evaluate the allegations.

3. I graduated from the University of Washington with a Bachelor of Arts degree in Comparative History of Ideas with an emphasis in computer science engineering and architecture design in 1997.

4. I filed my first set of (4) patents on October 13, 2000, at the age of 26; one of which has been recently issued. Since that first filing, I have filed about a dozen additional patents, which are all pending. True and correct copies of my first three patent applications, the fourth was filed as non-disclosable, a list of the additional patents pending and my recently issued patent are attached hereto as Exhibit 1.

5. In April, 2005, I formed MOD Systems, Inc. (formerly Pop Media) with Anthony Bay. Mr. Bay is not a software developer, he is a business person. Although MOD was formed with the intention to create a retail CD burning solution, MOD is now focused on providing retailers with a digital media solution (MP3files) to enable them to sell and distribute digital entertainment directly to consumers. When I was forced out of MOD in March of 2009, we had 42 employees. MOD, like Banana, is still in the development stage. By March 2007, I concluded that it was necessary to focus on flash memory and security protocols to secure the success of MetaWallet (described later). The same developments needed to occur in MOD systems; therefore I directed my efforts to establishing an investment and working relationship with Toshiba, the inventor and leading provider of flash memory. In October 2008, I was successful in obtaining $35,000,000.00 in "Series A"

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

financing from Toshiba Corporation and NRC. Mr. Arnold invested $3,500,000.00 in MOD for a 7.84% interest (prior to Toshiba financing).

6. Prior to receiving the Series A financing for MOD, the MOD books were scrutinized for 3 days by Morrison & Forester and Price Waterhouse Cooper for Toshiba. David Douglas, MOD's CFO, and Anthony Bay for the MOD board both had due diligence responsibilities as did Kyleen Cane who was brought on the Board in part to assist in the due diligence for the Series A and ongoing financial oversight. If any fraud or embezzlement had been found there is no way that Toshiba and NCR would have invested.

7. In June, 2006, I formed Banana Corporation, dba MetaWallet. Banana/MetaWallet was the culmination of my interest in creating a mobile payment system with technology. It is deep in design and development (with a unique patent filed and much research done) and the market potential is enormous; Diamond Management and Technology Consultants believes that the industry, mobile payments, has the potential to reach $60 billion in annual transactions. At the time of Mr. Arnold's investment in Banana (June, 2006) Bruggeman & Johnson Yeanopolis P.C. (BJY) prepared a *Valuation of Banana Corporation Business Plan* which established a fair market value of Banana Corporation with limited business models of one prepaid carrier per country and no remittance functionality, micro-transactions and other deficiencies to be $58,967,000 with 100% ownership. (See Declaration of Dennis M. Mandell, exhibit 2, C2, *Appraisal Report of Banana Corporation*, issuance date July 24, 2009, filed in conjunction with this declaration.) I approached Mr. Arnold to become an investor in the MetaWallet mobile banking system. Mr. Arnold signed Common Stock Purchase Agreements for $5,500,000 or 15.4% of the company's shares. Mr. Arnold and I are the only investors.

8. In March of 2007 I was introduced to Jan Wallace at a charity event where I was networking for private investors. My initial conversations with Ms. Wallace were

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

exclusively about her wealthy contacts, business-finance background and skills, and about alternative fundraising approaches.

9. We began a professional relationship. Ms. Wallace provided business advice and was able to introduce me to potential investors in my companies. After working closely together for some time we began a romantic relationship and continued to work together professionally. Jan wanted to own shares in Mod and to be on the Board but I told her that since we were involved personally, that it could not occur; that made her very unhappy.

10. The MOD Board of Directors began having discussions regarding the legal steps necessary to conclude the MOD Series A financing, per the Toshiba-MOD memorandum of understanding dated September 2007. Ms. Wallace suggested I meet with her friend Kyleen Cane, an attorney. So in June 2008 we flew to Los Angles and met with Ms. Cane regarding the legal needs of MOD. Prior to meeting Ms. Cane personally, I exchanged e-mails with her and her law firm, and provided her with information about both MOD and Banana, including business plans, formation documents, corporate structure and IP information. Shortly thereafter, I introduced her to Anthony Bay, the other MOD Board of Director's member, to mediate between us on how to proceed forward with financing the company and she was voted onto the Board of Directors in June of 2008. We invited her on to the Board and at the same time I signed an engagement agreement between MOD and herself to represent MOD as its business attorney. I trusted her and Jan completely. A true and correct copy of the engagement agreement dated June 25, 2008 between MOD and Kyleen Cane's law firm, Cane Clark LLP, is attached hereto as Exhibit 2.

11. In October 2008, Jan Wallace told me that she had heard that Mr. Arnold was investigating me and might be filing a lawsuit against me and that Jeff Smyth had contacted her. She said she was communicating with Smyth through her lawyer and that she was "protecting my interest." I later heard that Ms. Wallace had sought out Mr. Arnold, instigated the claim and provided him with material that she took from my computer. I do

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

not know exactly what she took, what she gave to Mr. Arnold or his lawyer or what the Demand Review Committee looked at but some of the allegations in the Complaint could only have come from Jan. For example, Complaint at paragraph F, 64-67, references a "Meteor Consulting" account in the Netherlands. This information could only have come from Jan or material she provided. In addition, the allegation that $15,000.00 to $25,000.00 per month has been or is being paid into the account is absolutely false. There is an account and the balance is, and always has been zero.

12. As I stated, I was introduced to Kyleen Cane by Jan, I put Kyleen on the MOD board in June of 2008 and prior to that time she advised me on several issues related to Banana and MOD. Also prior to that time Jan had physical access to my computers from which she would have taken the material. Both Jan and Kyleen also knew about the Toshiba deal for many months before it closed and they knew that there would be a great deal of money in MOD when the Toshiba/NCR transaction closed. Although the final papers were signed on October 17, 2008, the deal actually was finalized in Tokyo the month before and I had a Memorandum of Understanding in September 2007 and Jan and Kyleen knew this. I am certain that Jan took the material off of my computer prior to Kyleen joining the board. As I said, it was actually Jan who first told me that Arnold might be suing and then I started receiving e-mails that Jeff Smyth was investigating me so it is not disputable that she went to Arnold before that time.

13. The lawsuit was filed in February of 2009 and in March I was asked to resign from MOD by Kyleen Cane and Anthony Bay. I did so. Kyleen told me that it would be in my interest to name her as trustee of my MOD shares. Since then she has actively tried to force me out of the company. She demanded that I give up my entire interest in MOD to settle the Arnold lawsuit.

14. At the time I put Kyleen on the MOD board and gave my shares to her, she had provided me with a great deal of counsel and trusted her completely. Although I knew that

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

Jan and Kyleen had business dealings, I was unaware of the extensive professional relationship between Jan Wallace and Kyleen Cane and the only lawsuit against Jan that I knew about was one she lost when we were together. After I was forced out of MOD I began to do some research about Jan and Kyleen. I also hired a forensic investigator, Dennis Mandell. I pulled SEC filings off of www.sec.gov involving both Jan Wallace and Kyleen Cane and began to research lawsuits referenced therein. My research and that of my investigator have discovered that Jan and Ms. Cane have an extensive history of working together. I now know that Ms. Cane was the attorney for at least 6 companies that Jan was involved in as an officer or shareholder, including: MW Medical, MW Fitness, Davi Skin, Secured Diversified Investments, Dynamic Associates and Tele-Lawyer.

15. With regard to the allegations of financial impropriety by me, I will respond to in detail as this case progresses but during the past ten years as an entrepreneur, Board member, CEO, CTO, Chief Software, Hardware, Semi-conductor Architect, Eco-System Architect, and Design Architect I have made enormous personal contributions to Banana and MOD Systems Corporations and I would like the court to be aware of those.

16. I formed A DOT Corporation to serve as an incubation, research, and development company; it is here where I began the foundation of developing what I hope will be Banana's technology of payment systems, point of sale devices, digital security, and media delivery systems. My expertise in these areas led to the formation of it and the evaluation report from Bruggeman & Johnson Yeanopolis P.C. which calculated Banana to be worth $58,967,000 with 100% ownership, the subsequent contract with NuevaTel, the filing of patent titled "METHOD AND SYSTEM FOR VALUE TRANSFER BETWEEN MOBILE-PHONE USERS", and careful and full understanding of the micro-transaction problem. (See Declaration of Dennis M. Mandell, Exhibit 2, Y)

17. While I was the CTO of Fullplay, I worked specifically on digital media applications and created four patents; these patents have been evaluated by Brueggeman and Johnson

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

Yeanopolos, P.C. on October 26, 2007 to be worth $393,000,000 if pending and $494,000,000 if issued. (See Declaration of Dennis M. Mandell, exhibit 2, C1, October 29, 2007 report of the appraisal of certain patents pending). Prior to MOD obtaining the Series a financing it needed $5,000,000.00 to continue operations so Toshiba advanced the money but required a mortgage on patent applications owned by me. This agreement, leveraging my patents on behalf of MOD, this occurred on March 19, 2008. The MOD Board and MOD's corporate counsel Fenwick West were fully aware of this. As I mentioned earlier, Toshiba's investment was critical for the development of flash memory, the key to both the success of Banana and MOD and my participation secured the financing.

18. By the terms of an Asset Purchase Agreement dated July 1, 2005 (before Mr. Arnold's investment) A DOT sold to MOD the A DOT existing operations, including, but not limited to, the physical assets, employees, functional systems, as well as some limited intellectual property for $50,000. The actual fair value to MOD to acquire a seasoned technology development team, in an operational office, with existing contracts is invaluable. MOD essentially acquired a profitable, fully staffed and going company at a highly discounted price. A true and correct copy of the Asset Purchase Agreement dated July 1, 2005 is attached hereto as Exhibit 3.

19. Another unvalued contribution that I gave to MOD at the time I founded it was to allow virtually all of the revenues generated by A DOT's contracts with Microsoft to pass through to MOD, generating approximately $970,000 of revenue to MOD and profits estimated at $632,433. MOD needed these revenues for its survival from 2005 and 2006. I have never sought repayment of any of that money.

20. During the time of A DOT's rendering of services for MOD, MOD did not have any money to pay A DOT, or anyone else to pursue the concept or do the work. A DOT had to get the money to pay the third party contractors on its own. A DOT appropriately invoiced MOD, with those invoices going unpaid and resulting in an accumulated balance owing of

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

over $1,000,000.00. A DOT appropriately invoiced MOD and, thereafter MOD paid off that loan as a bona fide business expense.

21. In 2006 MOD contracted with A DOT to manufacture 75 PODs and 50 Store Appliances suitable for immediate demonstration, sale and deployment for Starbucks. A DOT directed the electrical and mechanical design of the devices making key design decisions, selecting components, buying components, managing the selection and oversight of prototyping vendors to fabricate and assemble the devices. A DOT did not charge for hardware design expertise on a NRE basis and did not charge an IP licensing agreement (5-20 percent is industry standard). If an outside vendor had taken upon the task, it would have cost the company at minimum $3,000,000; therefore A DOT will assign the savings of $2,000,000 in respect to fair and market values. It is due to A DOT's business relationships for manufacturing hardware design that MOD had its initial success.

22. I personally guaranteed MOD's office lease, telephone and open accounts. MOD could not qualify for credit so I took out a company card on my own credit. When MOD didn't have the financial resources to continue or to fund a new venture, I personally invested into MOD and/or loaned the company money to continue. Kenn Gordon and I worked at significantly reduced salaries part of the time so as to try and relieve the burden on MOD's varying, but limited financial resources.

23. Following the Series A agreement with MOD, I conducted a corporate audit to reconcile all my companies and put them in order with Derek de Bakker at Foster Pepper and Ron Braley at Lasher Holzapfel Sperry & Ebbeson. This good faith effort took extreme effort and reveals the extra steps I was taking to create order in company accounting and identify any questions surrounding IP. At the time, it was stated that these fees were non reimbursable business expenses due to the fact that I was being indemnified and protected by MOD; since I have not been protected by MOD, these fees of approximately $150,000 should be paid back to me.

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

24. I know now that Mr. Arnold's lawsuit was the result of material given to him and statements made to him by Jan Wallace. The allegations are false. Since the lawsuit was filed, Mr. Arnold, through his attorney, has constantly demanded payments from me personally including stock, real estate, a car and watches. He is not representing the interest of the company or the other shareholders.

25. I am the majority owner of MOD and Banana I object to any settlement. The Demand Review Committee comprised of Ms. Cane and Mr. Bay is not in any sense independent or impartial. I believe that Ms. Cane and Ms. Wallace set me up to take over and loot the company and if I am not put back in control they will destroy it.

DATED THIS 24 day of September, 2009.

Mark Phillips

MARK PHILLIPS

JOHNSON | FLORA

2505 Second Avenue, Suite 500
Seattle, WA 98121
(t) 206.386.5566 (f) 206.682.0675

Exhibit 1

US 20020045960A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2002/0045960 A1

Phillips et al. (43) Pub. Date: Apr. 18, 2002

(54) SYSTEM AND METHOD FOR MUSICAL PLAYLIST SELECTION IN A PORTABLE AUDIO DEVICE

(75) Inventors: Mark E. Phillips, Seattle, WA (US); Eric P. Gibbs, Seattle, WA (US)

Correspondence Address:
SEED INTELLECTUAL PROPERTY LAW GROUP PLLC
701 FIFTH AVE
SUITE 6300
SEATTLE, WA 98104-7092 (US)

(73) Assignee: Interactive Objects, Inc., Bellevue, WA

(21) Appl. No.: 09/975,748

(22) Filed: Oct. 10, 2001

Related U.S. Application Data

(63) Non-provisional of provisional application No. 60/240,766, filed on Oct. 13, 2000.

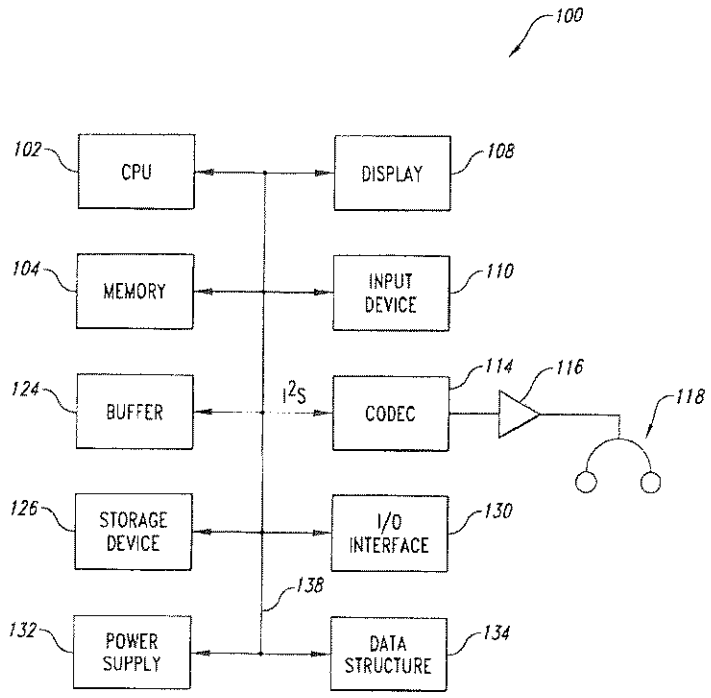
Publication Classification

(51) Int. Cl.[7] G06F 17/00; G10L 11/00

(52) U.S. Cl. 700/94; 381/61; 704/270

(57) ABSTRACT

A portable audio playing device implements a jukebox manager function to permit the simple generation of musical playlists and the alteration or editing of existing playlists. Data, such as MPEG-3 data or other conventional audio format data, may be readily downloaded into the system for storage in a solid state memory or in a spinning media device. The audio tracks are associated with one or more metatags that are used to describe the content of each track. The metatags and associated audio tracks are stored in a data structure that may be implemented as a database or other convenient data structure that readily permits searching by user-specified search terms. The user generates a new playlist by selecting one or more metatags corresponding to the desired musical tracks. The system queries the data structure using the user-specified metatags and automatically generates a playlist containing one or more audio tracks whose metatags correspond to the user-specified metatags. Alternatively, the system may perform the same query and simply generate a results list that will allow the user to manually specify which of the audio tracks identified by the search process will be added to the newly created playlist. The system also permits the simple editing of existing playlists. New audio tracks may be added in the manner described above using metatags for searching or maybe manually added from the list of stored audio tracks. The system readily supports different audio formats and different playlist types.



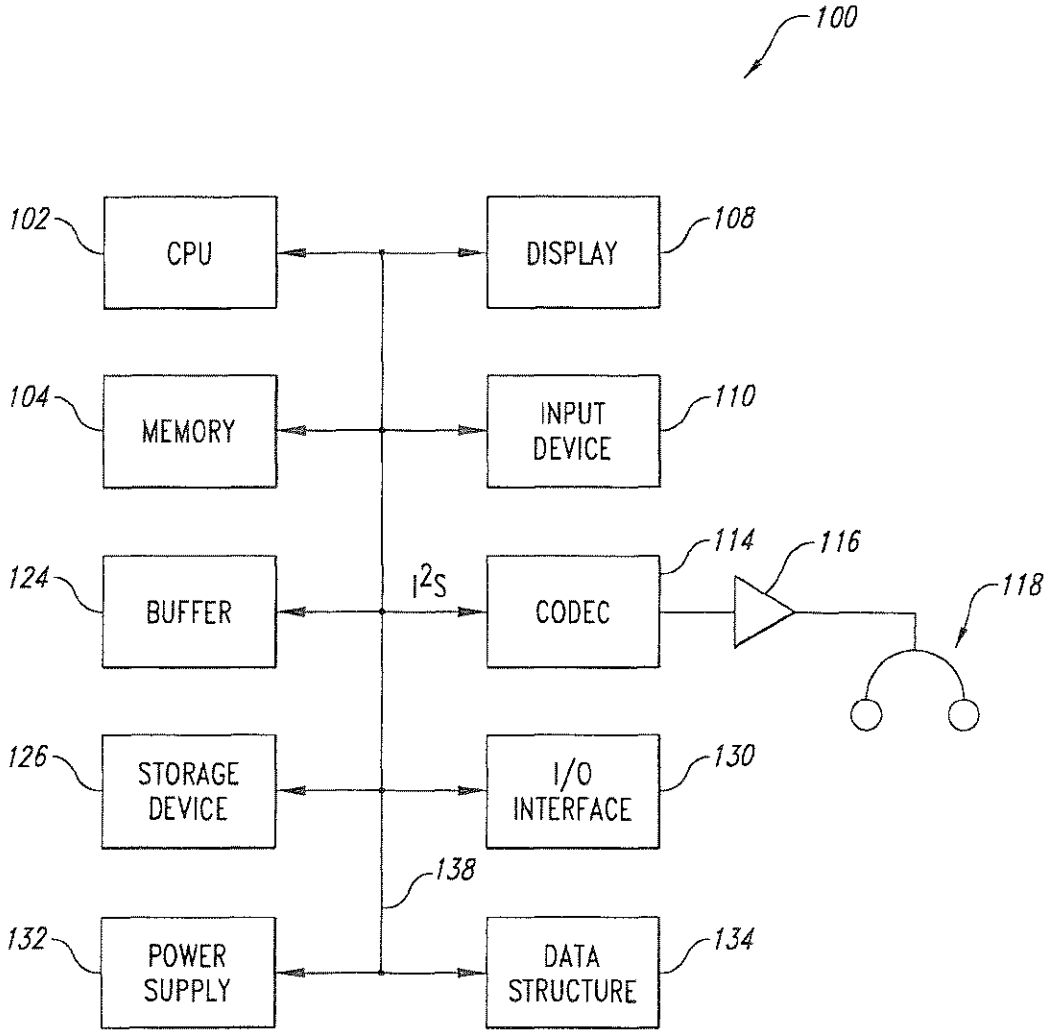

100
102
CPU
108
DISPLAY
104
MEMORY
110
INPUT DEVICE
124
BUFFER
I2S
114
116
CODEC
118
126
STORAGE DEVICE
130
I/O INTERFACE
138
132
POWER SUPPLY
134
DATA STRUCTURE

Fig. 1

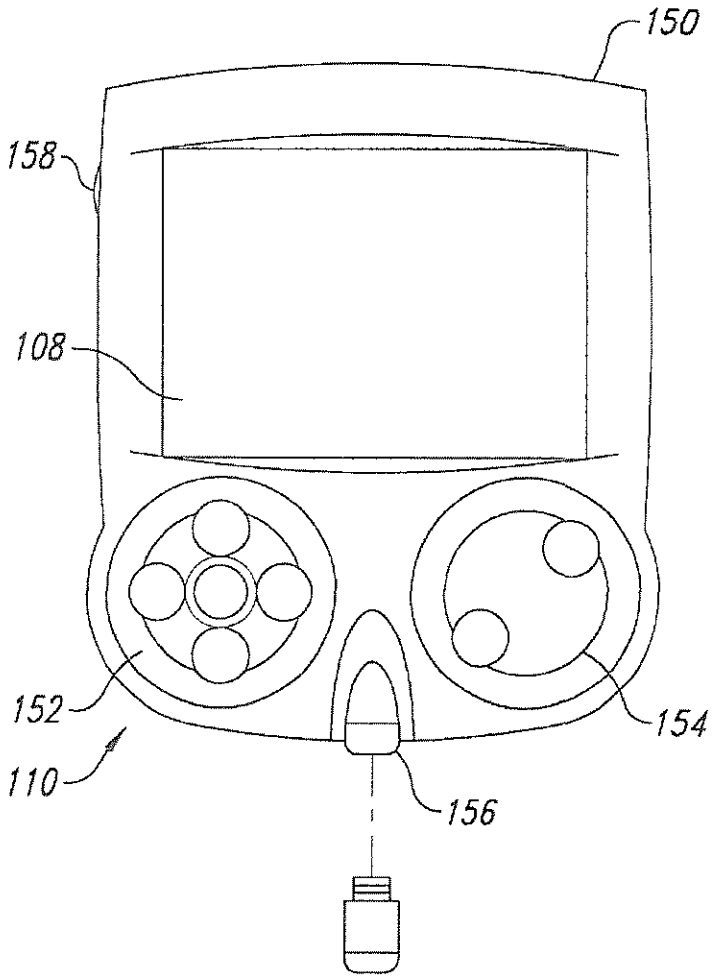


Fig. 2

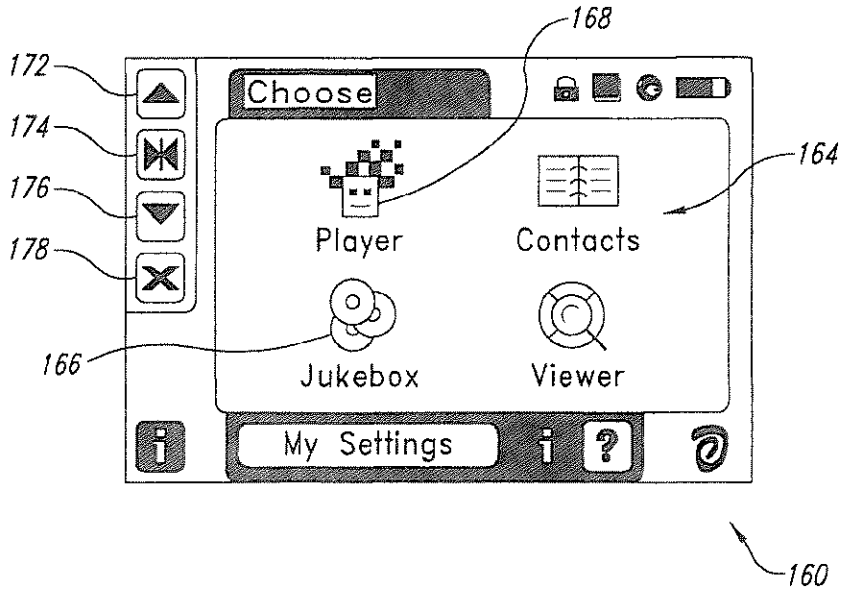

168
172
174
176
178
Choose
164
Player
Contacts
166
Jukebox
Viewer
My Settings
160

Fig. 3

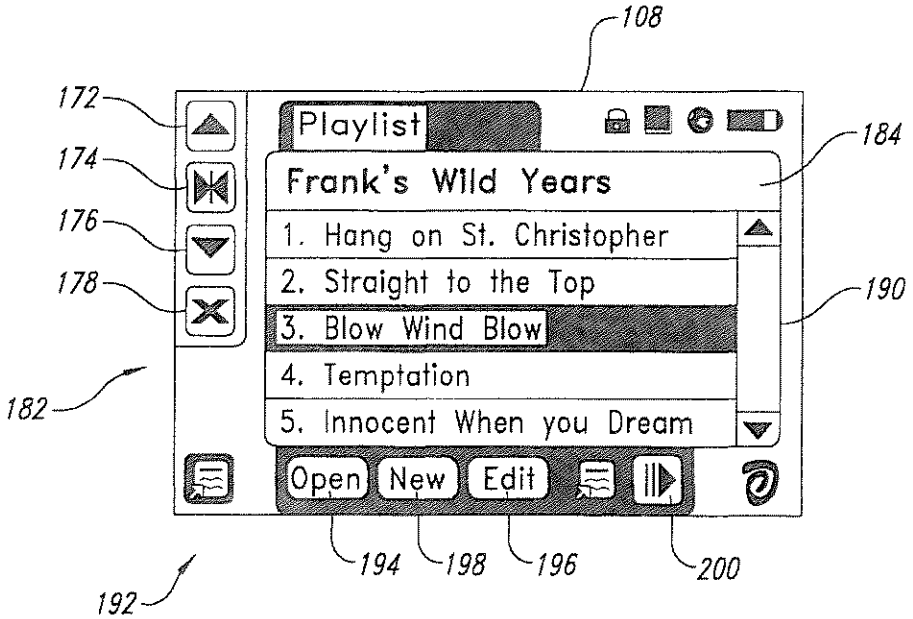

108
172
174
176
178
Playlist
184
Frank's Wild Years
1. Hang on St. Christopher
2. Straight to the Top
3. Blow Wind Blow
4. Temptation
5. Innocent When you Dream
190
182
Open
New
Edit
194
198
196
200
192

Fig. 4

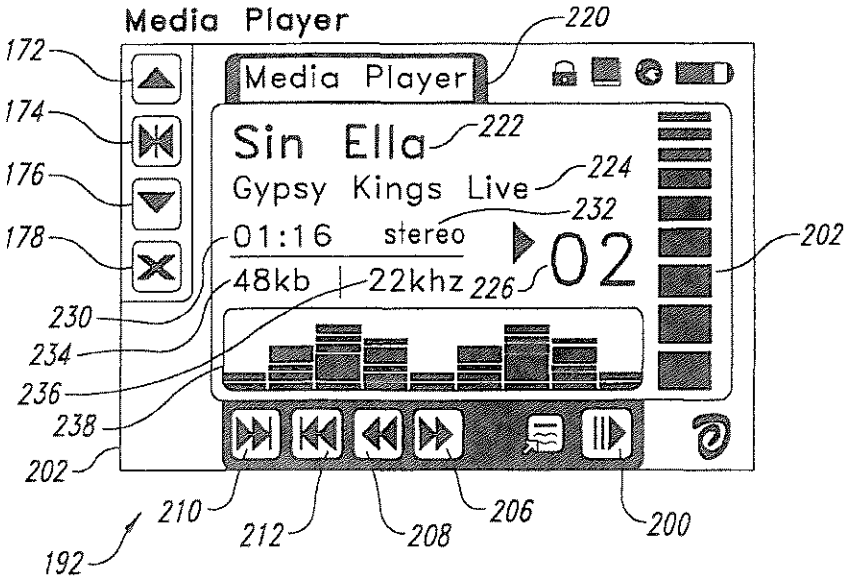

Media Player
172
174
176
178
220
Media Player
Sin Ella
222
Gypsy Kings Live
224
01:16 stereo
232
48kb | 22khz
226
02
202
230
234
236
238
202
210
212
208
206
200
192

Fig. 5

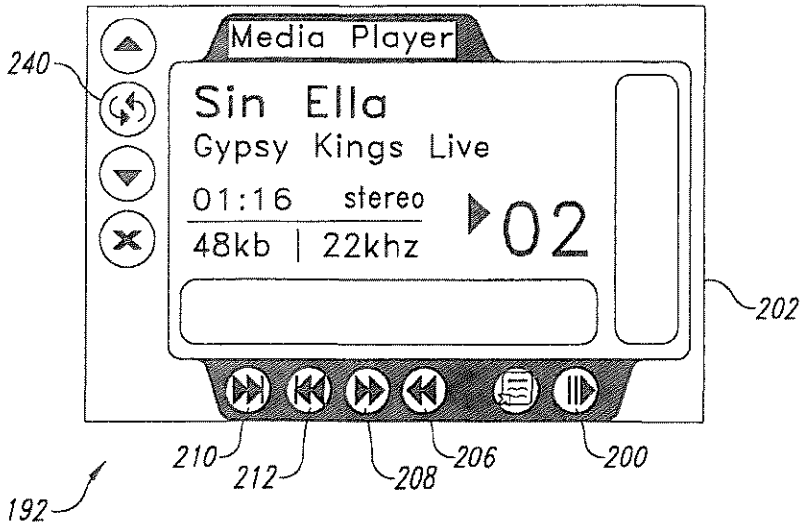

240
Media Player
Sin Ella
Gypsy Kings Live
01:16 stereo
48kb | 22khz
02
202
210
212
208
206
200
192

Fig. 6

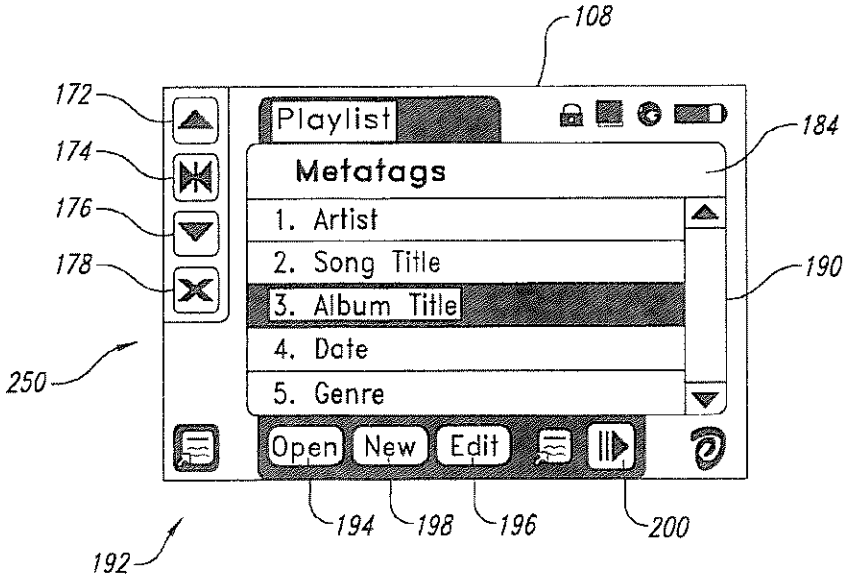

108
172
174
176
178
Playlist
184
Metatags
1. Artist
2. Song Title
3. Album Title
4. Date
5. Genre
190
250
Open
New
Edit
194
198
196
200
192

Fig. 7

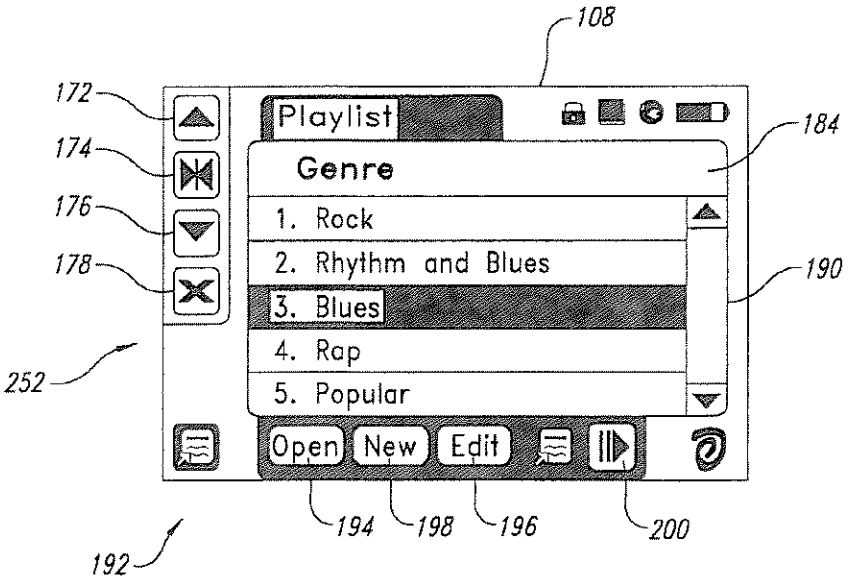

108
172
174
176
178
Playlist
184
Genre
1. Rock
2. Rhythm and Blues
3. Blues
4. Rap
5. Popular
190
252
Open
New
Edit
194
198
196
200
192

Fig. 8

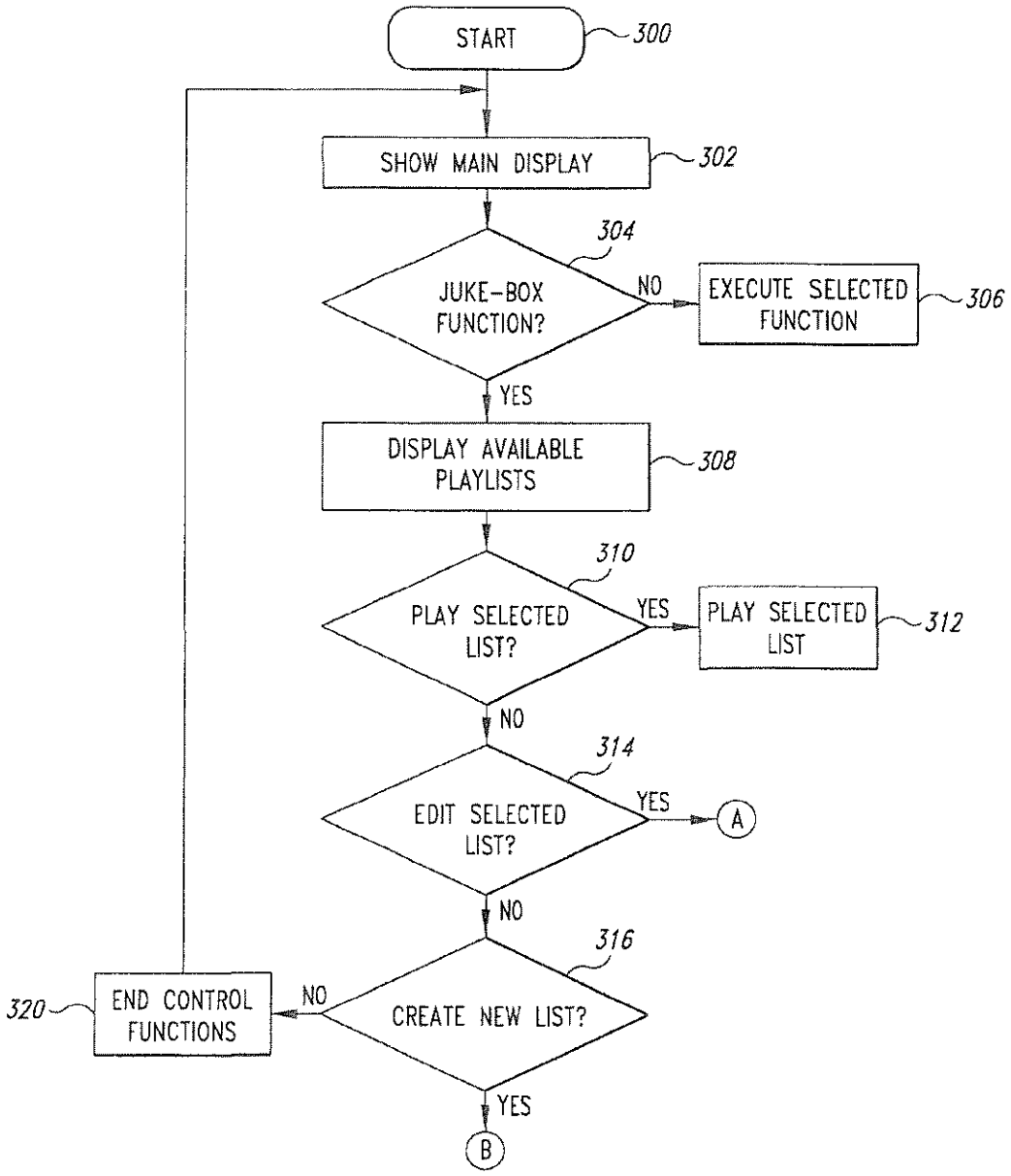

START
300
SHOW MAIN DISPLAY
302
304
JUKE-BOX FUNCTION?
NO
EXECUTE SELECTED FUNCTION
306
YES
DISPLAY AVAILABLE PLAYLISTS
308
310
PLAY SELECTED LIST?
YES
PLAY SELECTED LIST
312
NO
314
EDIT SELECTED LIST?
YES
A
NO
316
CREATE NEW LIST?
NO
320
END CONTROL FUNCTIONS
YES
B

Fig. 9

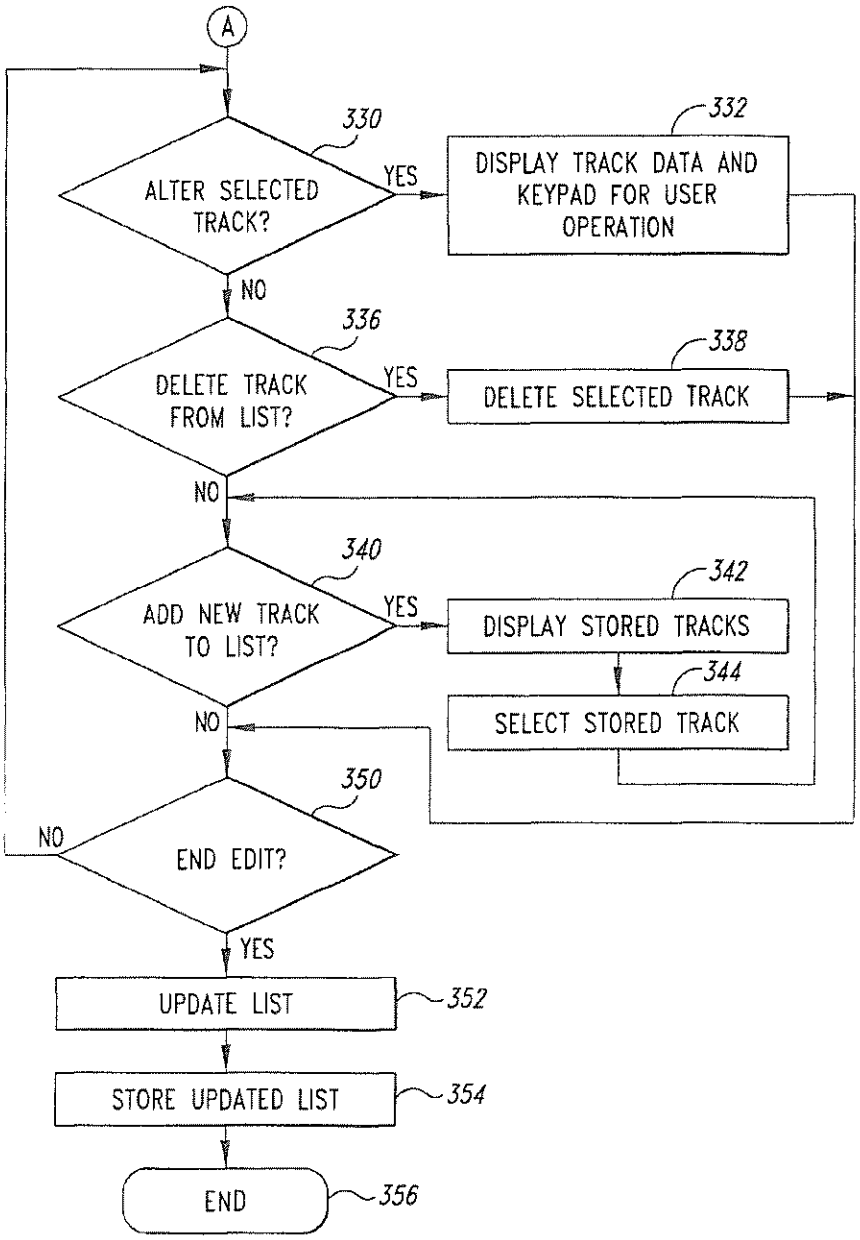

A
330
ALTER SELECTED TRACK?
YES
332
DISPLAY TRACK DATA AND KEYPAD FOR USER OPERATION
NO
336
DELETE TRACK FROM LIST?
YES
338
DELETE SELECTED TRACK
NO
340
ADD NEW TRACK TO LIST?
YES
342
DISPLAY STORED TRACKS
344
SELECT STORED TRACK
NO
350
END EDIT?
NO
YES
UPDATE LIST
352
STORE UPDATED LIST
354
END
356

Fig. 10

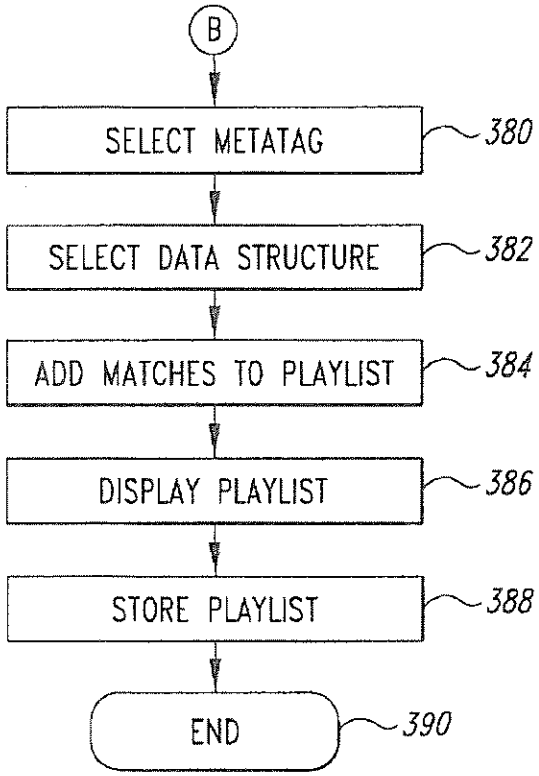

B
SELECT METATAG
380
SELECT DATA STRUCTURE
382
ADD MATCHES TO PLAYLIST
384
DISPLAY PLAYLIST
386
STORE PLAYLIST
388
END
390

Fig. 11

SYSTEM AND METHOD FOR MUSICAL PLAYLIST SELECTION IN A PORTABLE AUDIO DEVICE

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims the benefit of U.S. Provisional Patent Application No. 60/240,766 filed Oct. 13, 2000, where this provisional application is incorporated herein by reference in its entirety.

TECHNICAL FIELD

[0002] The present invention is related generally to portable audio devices and, more particularly, to a system and method for cataloging and creating playlists of audio data files.

BACKGROUND OF THE INVENTION

[0003] Portable audio devices have evolved from large cumbersome analog tape players to highly miniaturized digital storage devices. Early portable audio devices were typically in the form of analog tape players that sequentially played musical selections (or other audio presentations). For example, a prerecorded audio tape could be purchased by the user and sequentially played in a portable tape player. However, the user had no control over the sequence of play other than to stop the playing and manually fast forward or rewind to skip over one or more selections.

[0004] With the advent of portable digital devices in the form of compact disk (CD) players, the user has additional flexibility in the selections of songs from a CD. For example, some CD players permit the user to manually enter the sequence of musical tracks that will be played rather than play the musical tracks in a predetermined sequence from start to finish. Alternatively, some CD players also include a "random" mode in which musical tracks are randomly selected. However, the CD players described above are still limited to the selection of musical tracks on a single CD. Digital musical devices have been designed to eliminate all moving parts. These devices incorporate solid state memory storage technology and utilize digital processing capabilities, such as data compression, to minimize data storage requirements. A popular musical format, known as Motion Pictures Expert Group layer 3 (MPEG-2 layer 3) defines a digital musical format that plays "near-CD quality" music from a relatively small digital file as compared with the original digital file stored on a CD. Using known data compression techniques, the data structure defined by MPEG-2 layer 3, sometimes abbreviated as MP3, is approximately one tenth the size of a comparable data file on a CD.

[0005] With the introduction of large storage capacity MP3 players, the user may record and store a large number of musical data files. However, track selection and organization of such data files cannot be readily accomplished with conventional techniques. Therefore, it can be appreciated that there is a significant need for a system and method that will allow easy organization of data files in a portable digital audio device. The present invention provides this, and other advantages, as will be apparent from the following detailed description and accompanying figures.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] **FIG. 1** is a functional block diagram of an exemplary embodiment of the present invention.

[0007] **FIG. 2** is a top plan view of one embodiment of the present invention.

[0008] FIGS. **3-8** are various screen displays illustrating the operation of the present invention in various data entry and editing modes.

[0009] FIGS. **9-11** together form a flow chart illustrating the operation of the system of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0010] The present invention is directed to techniques that permit the user to implement a form of "jukebox" on a portable audio device. A jukebox manager allows the user to create, edit, and utilize a playlist comprising one or more data files. In a typical implementation, each data file is a separate musical track. The jukebox manager provides the user with a quick, flexible tool to organize music into playlists. As will be discussed in greater detail below, the system supports different play file formats and allows the user to generate or edit a playlist regardless of the format of the data file.

[0011] The present invention is embodied in a system **100**, illustrated in the functional block diagram of **FIG. 1**. The system **100** includes a central processing unit (CPU) **102** and a memory **104**. The CPU **102** may be implemented using a device, such as the ARM **7209** from Cirrus Logic or other processor designed for operation as an MP3 player. However, those skilled in the art will appreciate that the CPU **102** may be implemented using any convenient processor, such as a microprocessor, embedded controller, digital signal processor (DSP) or the like. The present invention is not limited by the specific form of the CPU **102**. The memory **104** may typically include both random access memory (RAM) and readonly memory (ROM). In one embodiment, the ROM portion of the memory **104** may be implemented using a flash program memory or a NAND flash memory. In addition, the memory **104** includes a basic input output system (BIOS), which contains instructions that allow the CPU **102** to communicate with various peripheral devices.

[0012] In addition, the system **100** includes a display **108**. In an exemplary embodiment, the display **108** is implemented as a liquid crystal display (LCD) to reduce overall power consumption. In one example, the display **108** may be a 240 by 160 pixel LCD subsystem, such as may be commercially purchased from a number of vendors. The display **108** may conveniently provide instructions to the user as well as programmable functions that may be context-sensitive. For example, when playing a music signal, the display **108** may provide commands associated with music playing, song information, and the like. For example, the display **108** may show the data sampling rate and number of kilobytes (Kb) in a particular data file. The display **108** may also include other information, such as power status, startup information, and the like.

[0013] The system **100** also includes an input device **110**. The input device **110** may be implemented as a series of electromechanical switches using conventional techniques. Alternatively, the input device **110** may be implemented in conjunction with the display **108** to provide a touch-sensitive display. A touch-sensitive display advantageously minimizes the need for electromechanical switches and further

provides labels on the display that may be readily altered to accommodate variations in the implementation of the system **100**. Alternatively, the input device **110** may comprise both electromechanical switches and a touch-sensitive display. Electromechanical switches and touch-sensitive displays are known in the art and need not be described in further detail herein. However, the present invention is not limited by the specific form of the input device **110**.

[0014] As those skilled in the art can appreciate, the data representing the audio signal is in the form of digital samples. The digital data must be converted to analog form to produce a useful signal for the user. The system **100** includes a coder/decoder (CODEC) **114**. The CODEC **114** is also sometimes referred to as a "compressor/decompressor" because the digital data samples are usually stored in a compressed form and are decompressed for playback. The CODEC **114** accepts a digital data stream and converts it to a representative analog signal. Different commercial CODECs are available for audio applications. Some CODECs, such as a code excited linear prediction (CELP) CODEC is designed for operations at relatively low frequencies and thus is particularly useful as a speech CODEC. Other forms of speech CODECs include adaptive delta modulation (ADM), pulse code modulation (PCM) and adaptive differential pulse code modulation (ADPCM).

[0015] Other forms of CODECs are designed for operation at higher data sampling rates and are thus useful for music applications. These music CODECs include MPEG or MP3 CODECs, G2 format, developed by Real Networks, Enhanced Perception Audio Decoder (ePAC), developed by Lucent, AC3 algorithm, which is a modified version of PCM, and Windows Media Audio (WMA), developed by the Microsoft Corporation. Some formats, such as the G2 format, may be used for both music and voice. Although the examples illustrated herein are directed to MP3 music format, those skilled in the art will recognize that the CODEC **114** illustrated in **FIG. 1** may be satisfactorily implemented using any of the known CODEC technologies for either speech applications, music applications, or both. Thus, the present invention is not limited by the specific implementation of the CODEC **114**.

[0016] In a typical embodiment, the system **100** may include multiple CODECs to process different file types. For example, an MP3 CODEC may be included to process music files encoded in accordance with an MP3 format. Other data files, such as audio books, may be processed using a different CODEC, such as the CELP CODEC. The playlists generated by the present invention may include data files of different types, such as an MP3 data file. The data file types may typically be identified by the file extension. For example, an MP3 file is followed by an extension ".m3u" while other data files may have a different file extension, such as ".1st." For the sake of convenience in operation, the system **100** of the present invention may display the names of data files without file extensions since multiple CODECs are available to process different file types. From the perspective of the end user, the file type and format is irrelevant so long as the user knows that the system can process the data file(s) selected by the user. Therefore, the playlist need not contain any file extensions. Alternatively, the system can display file extensions simply to provide the user with additional information concerning the various file types.

[0017] In an MP3 environment, the digital data is provided to the CODEC **114** using an I^2S bus. The I^2S bus is a high speed serial bus that is well known to those of ordinary skill in the art. As such, implementation details of the I^2S bus need not be provided herein. The CODEC **114** receives the data on the I^2S bus and converts it from digital data form to analog data. An analog amplifier **116** has an input terminal coupled to the output of the CODEC and receives the analog signal thereon. The amplifier **116** provides the necessary amplification and drive capability to power an audio output device **118**, such as a pair of headphones. It should be noted that in a typical implementation, the output of the amplifier **116** is coupled to a standard ⅛ inch phone jack (not shown). The headphones **118** plus into the phone jack.

[0018] The system **100** also includes a buffer **124** that receives and temporarily stores digital data and provides the digital data to the CODEC **114**. As will be discussed below, the buffer **124** receives data from a storage device **126**. The buffer **124** may be a stand-alone device, or may be a portion of the memory **104**. The use of the buffer **124** in optimizing the response of the storage device **126** will be discussed below.

[0019] The storage device **126** is typically implemented as a spinning media device, such as a micro-drive, click drive, or the like. The storage device **126** has a controllable motor (not shown) that is only enabled when the system **100** requires a data transfer to or from the storage media. The optimization of the storage device **126** includes a determination of when to start the motor on the storage device to allow it to come up to full speed, and how long to maintain power to the motor so as to transfer the desired amount of data from the storage media to the buffer **124**.

[0020] Those skilled in the art will recognize that the storage device **126** is an optional component and may be eliminated without adversely affecting the operation of the present invention. A number of portable audio devices contain no storage device **126**, but rely solely on the memory **104** to store the musical tracks. For the sake of completeness, the buffer **124** and storage device **126** are described herein. The buffer **124** is implemented in the system to optimize data transfer from the storage device **126**. Although it is beyond the scope of the present invention, the buffer **124** may be allocated into a large number of buffer portions with one of the buffer portions being actively used to transfer data to the CODEC **114** while the remaining buffer portions are available for data transfer from the storage device **126**. If the system **100** is implemented without the storage device **126**, the buffer **124** may also be eliminated without adversely affecting the operation of the system. In this implementation, the musical track data is transferred directly from the memory **104** to the CODEC **114**. Because the memory **114** is a solid state memory, data transfer rates are sufficiently high to accommodate satisfactory data transfer to the CODEC so as not to cause interruptions in the generation of output data.

[0021] The system **100** also may include an optional input/output (I/O) interface **130**. The system **100** may include any conventional form of I/O interface and may typically include a serial interface and/or a universal serial bus (USB) interface. The operation of a serial interface and USB interface are well-known in the art and need not be described in greater detail herein. Although illustrated as a

single I/O interface **130**, those skilled in the art will recognize that the I/O interface **130** is intended to illustrate the function of one or more conventional interfaces.

[**0022**] A power supply **132** provides power to all of the components of the system **100**. In an exemplary embodiment, the power supply **132** comprises two or more AAA batteries. A voltage regulator (not shown) in the power supply **132** provides a regulated voltage of approximately 3.1 VDC. The power supply **132** may also include provisions, such as an external power supply jack **170** (see **FIG. 2**), to permit the introduction of power from an external source, such as a cigarette lighter in an automobile, or the like.

[**0023**] The system also includes a data structure **134** to store data related to user-generated playlists and associated data. In one embodiment, the data structure **134** may be implemented as a database. However, those skilled in the art will recognize that any convenient form of known data structure will operate satisfactorily with system **100**. Furthermore, the data structure **134** may be a portion of the memory **104** or a stand-alone data storage element. The present invention is not limited by the specific form in which the data structure **134** is implemented.

[**0024**] The various components of the system **100** are coupled together by a bus system **138**. The bus system **138** may include a data bus, control bus, the I^2S bus, a memory bus, and the like. However, for the sake of simplicity, these various buses are illustrated in **FIG. 1** as the bus system **138**.

[**0025**] The system **100** is intended for portable operation. The various components described above are typically implemented as one or more integrated circuits on a printed circuit (PC) board (not shown). The PC board power supply **132**, display **108**, input device **110**, and other components of the system **100** are enclosed in a case or housing **150**, as illustrated in **FIG. 2**. As further illustrated in **FIG. 2**, the input device **110** comprises a four-button key pad assembly **152**, a two-button key pad assembly **154**, and an optional joystick **156**. The four-button key pad **152** may be conveniently configured to function in a manner similar to well-known hand-held electronic games. Alternatively, the four-button key pad **152** can be replaced with a membrane (not shown) to permit the operation of four hardware buttons in a manner similar to a top hat switch on a joystick wherein one or two of the buttons may be activated to provide eight unique switch settings. In yet another alternative, the four-button key pad **152** or the two-button key pad **154** could be replaced with a position-sensing membrane, such as a touch pad commonly used in laptop computers. Those skilled in the art will recognize that other configurations may also be used for the input device **110**. As will be described in greater detail below, the display **108** may conveniently comprise touch-sensitive display technology that will allow readily alterable configurations for control buttons that will correspond with the particular data shown on the display **108**. A power switch **158** may be conveniently installed in the side of the housing **150** to allow the user to turn the system on and off.

[**0026**] When power is first applied to the system **100**, the display **108** may be configured to illustrate a main menu, such as illustrated in the screen display **160** of **FIG. 3**. The screen display **160** may include a series of icons **164**, such as a jukebox icon **166**, a player icon **168**, and the like. In addition to icons **164**, the screen display **160** may include touch-sensitive programmable controls, such as a "Scroll Up" control button **172**, a "Selection" control button **174**, a "Scroll Down" control button **176** and an "Exit" control button **178**. The operation of a touch-sensitive screen to implement these buttons are well known and need not to be described in any greater detail herein. Furthermore, the operation of the buttons, such as the Scroll Up button **172** and the Scroll Down button **176** are well known in the art and need not be described in detail. Activating the Scroll Up button **172** or the Scroll Down button **176** will cause the display to highlight a different one of the icons **164**. When the desired icon is highlighted, such as by reverse video or other conventional technique, the user may activate the selection button **174** to activate the selected function.

[**0027**] **FIG. 4** illustrates a sample screen display **182** shown by the system in response to the activation of the jukebox icon **166** and the selection of one playlist. As previously noted, the system **100** supports a plurality of different playlists. The screen display **182** comprises a playlist title portion for a playlist title display **184** to permit the user to readily identify the selected playlist. The user may simply activate the playlist to play musical tracks in the predetermined sequence shown in the playlist by pressing the Selection control button **174**. When a display list is first shown on the display **108**, the first entry in the playlist may be automatically selected and indicated using, by way of example, reverse video. The user may also scroll through the selected playlist using a scroll bar **190** in a well-known fashion or, alternatively, simply by touching the touch-sensitive display **108** at a point corresponding to the desired musical track. The system **100** may also be configured to allow the user to scroll through the selected playlist using the Scroll Up button **172**, a Scroll Down button **176**, and the Selection control button **174** in the manner described above to select a musical track out of the sequence illustrated in the playlist.

[**0028**] The user may also control the operation of the system **100** to open or edit playlists, or create new playlists using additional programmable control buttons **192** on a predetermined portion of the touch-sensitive display **108**. The Programmable control buttons **192** may comprise buttons such as a "Open" control button **194**, an "Edit" control button **196** and a "New" control button **198**. The Open control button **194** may be used to display a number of different playlists and permit the user to select from one of the displayed playlists in the manner described above. That is, the user may activate the scroll bar **190** or the Scroll Up button **172**, the Scroll Down button **174**, and the like, to navigate through the displayed playlists. As the displayed playlists scroll up or down the display **108**, a selected display list is shown in a highlighted fashion, such as reverse video. The user opens the selected playlist using the Selection control button **174** or another one of the convenient Programmable control buttons **192**. The user may edit a selected playlist by selecting the Edit control button **196**.

[**0029**] The user may edit an existing playlist by activating the Edit control button **196**. Activation of the Edit control button **196** will cause the system **100** to display the names of already established playlists. The user may manipulate through the lists of playlists using, by way of example, the scroll bar **190** to select the desired playlist. When the desired playlist has been selected, the display **108** will indicate the

musical tracks already selected in the playlist, as illustrated in **FIG. 4**. In an exemplary embodiment, the first musical track in the playlist is highlighted using, by way of example, reverse video. The user selects a particular musical track in the manner described above. The user can edit a selected musical track, to correct misspellings or other information, delete an existing musical track from the current playlist, or add additional musical tracks to the selected playlist using conventional editing techniques. The user exits the edit mode by activating the Exit control button **178**.

[0030] In addition to editing an existing playlist, the user may elect to create a new playlist by activating the New control button **198**. When the user activates the New control button **198**, the display **108** may be configured to show all musical tracks currently stored in the memory **104**. The user may scroll through the list of musical tracks using conventional controls, such as the scroll bar **190**. As the user scrolls through the list of musical tracks, a selected musical track may be highlighted using, by way of example, reverse video. Other conventional techniques, such as bold video, underlined text, an asterisk or other indicator, may also be used to indicate the selected musical track. To enter a selected musical track into the new playlist, the user may activate the Selection control button **174**. The user may scroll through the displayed list of stored musical tracks and select other musical tracks in the manner described above to thereby enter them into the playlist. When the playlist is completed, the user may exit the data entry mode by selecting the Exit control button **178**. Thus, the system **100** has provided the user with a simple technique for creating music playlists.

[0031] When a playlist or individual musical track has been selected, that selection may be played by activating the Selection control button **174** or a special control button, such as a "Play/Pause" button **200**. When a selected musical track begins to play, the touch-sensitive display **108** may be reprogrammed to show a screen display **202**, illustrated in **FIG. 5**. The touch-sensitive display **108** has also been changed such that the control buttons perform different functions relevant to a media player. For example, the Scroll Up control button **172** and Scroll Down control button **174** may now be used to control the volume. A graphical representation **204** may provide visual cues to the user as to the volume level. The programmable control buttons **192** may now comprise a Fast Forward button **206** and Rewind button **208** to advance or rewind within the selected musical track. A Skip Forward button **210** may be used to automatically advance to the next musical track in the playlist while a Skip Rewind button **212** may be activated to rewind to the beginning of the current musical track if activated once and rewound to the beginning of the previous musical track in the playlist if activated twice within a short period of time. In addition, the Play/Pause control button **200** may be used in the manner previously described.

[0032] In addition to control buttons, the display screen **202** can provide user information, such as the currently selected function **220**, a title **222**, an artist name **224**, and a track selection **226**. Other information, such as an elapsed time **230**, stereo indicator **232**, sample rate indicator **234**, and bandwidth indicator **236** may also be provided on the display screen **202**. In addition, an exemplary embodiment of the system **100** may include a graphical equalization display **238** to indicate the relative power of signals at different frequency bands. Those skilled in the art will recognize that numerous variations are possible with the present invention. For example, the graphical equalization display **238** can be eliminated and replaced with other information, such as metatags indicating categories or other identifier tags that correspond to the selected musical track.

[0033] One convenient aspect of on-screen programming using the display **108** is that many configurations are possible. An alternative configuration of the media player is illustrated in **FIG. 6** where the programmable controls **192** have a different appearance, but perform the same functions as previously described with respect to **FIG. 5**. In addition, the Scroll Up control button **172**, Scroll Down control button **176** and Exit button **178** have a different appearance in the display screen **202** of **FIG. 6**, but perform identical functions to those described above with respect to the corresponding buttons in **FIG. 5**. In **FIG. 6**, the selection control button **174** has been replaced with a Repeat control button **240** to permit the user to repeat a selected musical track or selected musical playlist. Other programmable features, such as random selection of musical tracks within a playlist, and the like may also be readily provided using the touch-sensitive display **108**.

[0034] Although the operation of the system **100** has been described with respect to buttons on the touch-sensitive display **108**, similar control of the system may be accomplished using, by way of example, the four-button key pad **152** (see **FIG. 2**) and the two-button key pad **154**. Essentially, the buttons of the four-button key pad **152** and two-button key pad **154** are mapped into the functions described above with respect to the Programmable control buttons **192** and the control buttons **172-178**. The operation of the four-button key pad **152** and two-button key pad **154** is within the scope of knowledge of one of ordinary skill in the art and thus, need not be described in greater detail herein.

[0035] The operation of the system **100** to open, edit, or create playlists has been previously described. In addition to selection of musical tracks by title, the system **100** advantageously allows the selection of musical tracks using metatags. In an exemplary embodiment, the system **100** creates the data structure **134** (see **FIG. 1**) to store metatags corresponding to musical tracks stored in the memory **104** (see **FIG. 1**). The data structure or database **134** may be part of the memory **104** (see **FIG. 1**) or a separate data storage element. Those skilled in the art will recognize that any one of a number of well-known data structures may be satisfactorily used to implement the data structure described herein. For the sake of convenience, the data structure **134** will be subsequently described as a database. However, the present invention is not limited by the specific implementation of a data structure to store metatags.

[0036] A number of different data elements may be used as metatags. For example, the artist's name, song title, album title, date, copyright, or any other information associated with a musical track can be potentially used as a metatag. In an exemplary implementation, the user may elect to create a new playlist by activating the New control button **198** (see **FIG. 4**) using metatags to describe the desired musical tracks. In this example, illustrated in **FIG. 7**, the display **108** shows a screen display **250** that lists a series of possible metatags for selection by the user. In an exemplary embodiment, the first metatag in the list of metatags is automatically

selected. The user may scroll through the list using, by way of example, the scroll bar **190** to select a desired metatag, as illustrated in **FIG. 7**. As noted above, the system **100** can automatically generate a playlist based on the user-selected metatag or provide a list of musical tracks that match the selected metatag for display and subsequent manual selection by the user. For example, if the user selected the metatag "Artist," the system **100** would permit the user to enter the name of a desired artist or, alternatively, will display the artist name for all musical tracks stored in the memory **104** (see **FIG. 1**). When the user selects a desired artist, the system may automatically generate the playlist and include all songs stored in the memory **104** that have a metatag corresponding to the user-selected artist name. Alternatively, the system **100** can display all musical tracks whose metatag corresponds to the user-selected artist name and thereby permit the user to manually select which musical tracks will be added to the playlist.

[0037] In addition to the metatags discussed above, other metatags, such as musical genre may be used as a metatag. For example, songs may be classified as "Rock,""Blues, ""Rap," and the like. If the user selects a particular metatag, the system **100** accesses the database to determine which musical tracks stored in the memory **104** (see **FIG. 1**) correspond to the selected metatag. If the user selects genre as the desired metatag, the system **100** may generate a screen display **252** on the display **108**, as illustrated in **FIG. 8**, to list the various musical genre for musical tracks stored in the memory **104**. As noted above, the first item in the list may be automatically selected and the user may alter the selection using, by way of example, the scroll bar **190**. In the example illustrated in **FIG. 8**, the user-selected musical genre is "Blues." The user may activate the selection using the Selection control button **174**. Once a particular genre, such as Blues, has been selected, the system **100** may search the data structure **134** (see **FIG. 1**) and automatically generate a playlist containing the musical tracks stored in the memory **104** whose metatags match the selected musical genre (i.e., Blues). Alternatively, the system **100** may search the data structure **134** and create a list of all musical titles stored in the memory **104** whose metatag matches the selected musical genre. The list may be shown on the display **108** to permit subsequent manual selection by the user.

[0038] It should be noted that each musical track may have a number of different metatags to easily enable the user to search the data structure and automatically generate playlists. The association of musical tracks with multiple metatags makes it easier for the user to search for desired musical tracks. In certain cases, a musical track may appear in more than one category. For example, certain musical tracks may be considered to belong to multiple genre, such as "Rock" and "Popular."

[0039] In an alternative embodiment, the system **100** permits searching by multiple metatags. For example, the user may wish to search the data structure **134** for musical tracks that match metatags for both artist name and a particular date. In another example, the user may wish to select a particular musical genre, such as "Rock" and date to automatically generate a musical playlist of rock songs prior to a user-selected date.

[0040] The operation of the invention is illustrated in the flowchart of FIGS. **9-11**. At a start **300**, illustrated in **FIG. 9**, it is assumed that the system is under power or has just been turned on by the user. In step **302**, the system **100** shows the main display, such as illustrated in **FIG. 3**. In decision **304**, the system determined whether the user has selected the jukebox function. If the user has not selected the jukebox function, the result of decision **304** is NO. In that event, the system moves to step **306** and executes the selected function, such as displaying a contact list of user-entered names, addresses and telephone numbers. These additional functions are beyond the scope of the present invention and will not be discussed in greater detail herein.

[0041] If the user has selected the jukebox function, the result of decision **304** is YES. In that event, the system **100** queries the data structure **134** and extracts the titles of all existing playlists and, in step **308**, the existing playlists are shown on the display **108** (see **FIG. 1**). In decision **310**, the system **100** determines whether the user has activated one or more buttons to select a playlist. If the user has selected a playlist for play, the result of decision **310** is YES and, in step **312**, the system plays the selected playlist by transferring data from the buffer **124** (or the memory **104**) to the CODEC **114** in a conventional fashion. As previously noted, the musical tracks of the selected playlist may be played sequentially in the sequence originally specified by the user when creating the playlist, in a new sequence specified by the user at the present time, or in some other fashion, such as random selection.

[0042] If the user has not selected a playlist to play, the result of decision **310** is NO. In that event, in decision **314**, the system **100** determines whether the user has selected a playlist for editing. If the user has selected a playlist for editing, the result of decision **314** is YES and the system enters an edit mode, described in the flowchart of **FIG. 10**. If the user has not selected a playlist for editing, the result of decision **314** is NO. In that event, the system determines, in decision **316**, whether the user has activated one or more buttons to create a new playlist. If the user has activated one or more buttons on the system **100** to create a new playlist, the result of decision **316** is YES and, the system enters a data entry mode illustrated in **FIG. 11**. If the user has not elected to create a new playlist, the result of decision **316** is NO and, in step **320**, the system ends the control function operation and, in one example, may return to display the main menu in step **302**. Those skilled in the art will recognize that a number of different possible flowcharts may be implemented by the present system. For example, the system **100** may return to decision **310** until the user selects an operation. In addition, the activation of other buttons, such as a main menu button (not shown) may be used to exit the control function process and return to the main display in step **302**. The flowchart of FIGS. **9-11** are intended simply as an illustration of possible control flow to create, edit, and play selected playlists. The present invention is not limited to the specific processing sequence illustrated in the flowcharts of FIGS. **9-11**.

[0043] As previously stated, the user may activate one or more of the buttons on the system **100** to edit a selected playlist. If the user has elected to edit a selected playlist, the result of decision **314** in **FIG. 9** is YES. In that event, the system **100** moves to decision **330**, illustrated in **FIG. 10**, to determine whether the user has elected to alter a selected track. If the user has elected to alter a selected track, the result of decision **330** is YES. In step **332**, the system

displays stored data about the selected track and may further display a keypad (not shown) for user operation to change selected data. For example, the user may wish to edit the title of a musical track to correct a typographical error from a previous entry. The user can highlight the selected data element (e.g., the title) and activate the edit control button **196** (see **FIG. 4**). The user can operate the touch-sensitive display **108** to enter a new title. The altered data will be displayed and stored in subsequent steps described below.

[**0044**] If the user has not elected to alter a selected track, the result of decision **330** is NO. In that event, the system **100** moves to decision **336** to determine whether the user has activated one or more keys to delete a selected track from the playlist. If the user has elected to delete a track from the playlist, the result of decision **336** is YES. In that event, in step **338**, the system **100** deletes the selected track and the newly edited playlist is updated and stored in steps described below. The system **100** also checks to see if the user wishes to perform more edits, as will be described in greater detail below. If the user has not activated one or more buttons on the system **100** to delete a musical track from the playlist, the result of decision **336** is NO.

[**0045**] In decision **340**, the system **100** determines whether the user has activated one or more buttons on the system **100** to add a new musical track to an existing playlist. If the user has elected to add a new musical track to the playlist, the result of decision **340** is YES. In that event, in step **342**, the system **100** displays a list of all musical tracks that may be stored in the memory **104** (or the optional storage device **126**). In step **344**, the user selects the desired musical track to the selected playlist in the manner described above. In an exemplary embodiment, a musical track that may be stored on the optional storage device **126** may be relocated to the memory **104**. Following the selection of the stored musical track in step **344**, the system **100** returns to decision **340** to determine whether additional new tracks will be added to the selected playlist.

[**0046**] If no additional musical tracks are to be added to the existing playlist, the result of decision **340** is NO and the edit operation. Following the completion of the selected edit operation, such as altering the selected track in step **332**, deleting a selected track in step **338**, or adding selected tracks in steps **342-344**, the system **100** moves to decision **350** to determine if the user wishes to perform additional edit operations on the selected existing playlist. If the user does not wish to end the current editing session, the result of decision **350** is NO and the system may return to decision **330** to permit additional editing of one or more tracks in the existing playlist.

[**0047**] If the user wishes to end the editing session by activating, by way of example, the Exit control button **178** (see **FIG. 4**), the result of decision **350** is YES. In that event, in step **352**, the system **100** updates the existing playlist to include all edits performed by the user and, in step **354**, the system stores the newly edited playlist. As previously discussed, the edited playlists may be conveniently stored as part of the data structure **134**. The edit operation ends at **356**.

[**0048**] Returning momentarily to the flow chart of **FIG. 9**, if the user wishes to create a new playlist, the result of decision **316** is YES. In that event, the system executes processes illustrated in the flowchart of **FIG. 11** to create a new playlist. As previously discussed, the system **100** may simply display the titles of all musical tracks stored in the memory **104** and allow the user to manually select ones of the displayed musical tracks to add to the newly created playlist. **FIG. 11** illustrates the operation of the system **100** to generate a playlist using metatags. In step **380**, the user selects a desired metatag from the list shown, by way of example, in the screen display **250**, illustrated in **FIG. 7**. The user may select a metatag, such as genre, which causes the system **100** to display the display screen **252** listing the various genre metatags corresponding to the various musical tracks stored in the memory **104** (or in the storage device **126**). In addition, as noted above, the user may select more than one metatag to further refine the selection of musical tracks. Thus, step **380** may represent multi-step processes in which one or more screen displays are provided to the user to guide the user through the metatag selection process.

[**0049**] After one or more metatags have been selected in step **380**, the system **100** searches the data structure **134** (see **FIG. 1**) in step **382**. In one example implementation, the data structure **134** may be a conventional database in which search terms, such as the selected metatags, are provided as inputs to the database and results are produced by the database in the form of one or more musical tracks whose metatags correspond to the user-selected metatags.

[**0050**] In step **384**, the system automatically adds to the playlist musical tracks whose metatags match the user-selected metatags. The automatically selected playlist is displayed for the user in step **386**. The user may manually edit one or more of the musical tracks on the newly generated playlist in the manner described above with respect to the flowchart of **FIG. 10**. Alternatively, the system **100** may simply display the resultant matches and permit the user to manually select which musical tracks will be added to the newly created playlist. In step **388**, the completed playlist is stored in the memory **104** or, alternatively, in the data structure **134**. The process ends at **390**.

[**0051**] Thus, the system **100** provides a powerful but simple interface that allows the user to quickly generate playlists from stored musical tracks using one or more user-selected metatags. The system further provides simple editing processes that allow the user to readily alter existing playlists.

[**0052**] From the foregoing it will be appreciated that, although specific embodiments of the invention have been described herein for purposes of illustration, various modifications may be made without deviating from the spirit and scope of the invention. For example, the operation of the system **100** has been described using the example of musical tracks as the audio data files that are selected by a user and placed in playlists. However, the system **100** is applicable to any type of audio data file, such as audio books, as well as musical data files. Accordingly, the invention is not limited except as by the appended claims.

What is claimed is:

1. A system for the display and control of music selection in a hand-held portable multi-media device, the system comprising:

a housing sized to be held by a user;

a circuit board within the housing;

a battery power supply to provide electrical power to electrical circuitry on the circuit board;

a data structure to store a plurality of music data files, each music selection data file having identification data associated therewith;

a display to display data comprising a playlist indicating music data files to be played;

an input device operable by the user to select identification data associated with desired music data files for the playlist;

a processor responsive to the input device to select the music data files for the playlist based on the user selected identification data;

a CODEC to receive the selected music data files and convert the selected music data files to audio data; and

an audio output driver coupled to the CODEC to receive the audio data therefrom, the audio output driver further having an output and providing analog signals to the output for connection to an audio output device.

2. The system of claim 1 wherein the data structure contains music data files having different data format types.

3. The system of claim 1 wherein the data associated with the stored music data files comprises song names and the display displays the song names, the user manually generating the playlist by operating the user input device to select song names and the processor generating the playlist based on the selected song names.

4. The system of claim 1 wherein the data associated with the stored music data files comprises metatags and the display displays the metatags, the user generating the playlist by operating the user input device to select metatags and the processor generating the playlist based on the selected metatags.

5. The system of claim 1, further comprising an associated data structure wherein the associated data comprises a plurality of data types, the processor analyzing the music data file to determine one or more associated data types and storing each of the data types for each music data file in the associated data structure in association with the music data file.

6. The system of claim 5 wherein the processor selects the music data files for the playlist by generating an indicator to indicate a storage location in the associated data structure for an associated data type for each of the selected music data files.

7. The system of claim 1 wherein the associated data comprises a plurality of data types and the user selects a desired data type using the user input device, the display displaying the user-selected data type associated with each of the plurality of music data files.

8. The system of claim 1 wherein the associated data comprises a plurality of data types and the display displays all associated data types for a user-selected one of the music data files.

9. The system of claim 1, further comprising a selection data structure wherein the playlist is stored for subsequent use.

10. The system of claim 1 wherein the processor alters the stored playlist and wherein the altered playlist is stored for subsequent use.

11. The system of claim 1 wherein the processor is responsive to the input device to select music data files based on user-selection of a plurality of identification data associated with the music data files.

12. A method for the automatic control of music selection in a hand-held portable multi-media device, the method comprising:

storing a plurality of music data files, each music selection data file having identification data associated therewith;

sensing user operation of an input device to select identification data associated with desired music data files for the playlist;

selecting a portion of the music data files to generate the playlist based on the user selected identification data;

processing the selected music data files with a CODEC to convert the selected music data files to audio data; and

providing the audio data to an output for connection to an audio output device.

13. The method of claim 12 wherein the music data files have different data format types.

14. The method of claim 12 wherein the data associated with the stored music data files comprises song names, the method further comprising displaying the song names and sensing user-operation of the input device to manually generate the playlist by operating the user input device to select song names wherein selecting comprises generating the playlist based on the selected song names.

15. The method of claim 12 wherein the data associated with the stored music data files comprises metatags, the method further comprising displaying the metatags and sensing user-operation of the input device to select metatags wherein selecting comprises generating the playlist based on the selected metatags.

16. The method of claim 12 wherein the associated identification data comprises a plurality of data types, the method further comprising analyzing the music data file to determine one or more associated data types and storing each of the data types for each music data file in association with the music data file.

17. The method of claim 12, further comprising sensing user input to select a plurality of identification data wherein selecting music data files is based on the user-selected plurality of identification data associated with the music data files.

18. A computer-readable media that causes a processor to control of music selection in a hand-held portable multi-media device by performing the steps of:

storing a plurality of music data files, each music selection data file having identification data associated therewith;

sensing user operation of an input device to select identification data associated with desired music data files for the playlist;

selecting a portion of the music data files to generate the playlist based on the user selected identification data;

processing the selected music data files with a CODEC to convert the selected music data files to audio data; and

providing the audio data to an output for connection to an audio output device.

19. The computer-readable media of claim 18 wherein the data associated with the stored music data files comprises metatags, the computer-readable media causing the processor to perform the steps of displaying the metatags and sensing user-operation of the input device to select metatags wherein selecting comprises generating the playlist based on the selected metatags.

20. The computer-readable media of claim 18, further causing the processor to sense user input to select a plurality of identification data and select music data files based on the user-selected plurality of identification data associated with the music data files.

* * * * *