Exhibit 10
(Submitted Under Seal)

CONTAINS INFORMATION DESIGNATED AS SAMSUNG HIGHLY CONFIDENTIAL ATTORNEY'S EYES ONLY

UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA
SAN JOSE DIVISION

APPLE INC., a California corporation, Plaintiffs, vs. SAMSUNG ELECTRONICS CO., LTD., a Korean business entity, SAMSUNG ELECTRONICS AMERICA, INC., a New York corporation, and SAMSUNG TELECOMMUNICATIONS AMERICA, LLC, a Delaware limited liability company, Defendants. SAMSUNG ELECTRONICS CO., LTD., a Korean business entity, SAMSUNG ELECTRONICS AMERICA, INC., a New York corporation, and SAMSUNG TELECOMMUNICATIONS AMERICA, LLC, a Delaware limited liability company Counterclaim-Plaintiff, v. APPLE INC., a California corporation, Counterclaim-Defendants.	Civil Action No. 11-CV-01846-LHK

Expert Report of Tony D. Givargis, Ph.D.
Regarding Invalidity of the Asserted Claims of U.S. Patent No. 7,698,711

CONTAINS INFORMATION DESIGNATED AS SAMSUNG HIGHLY CONFIDENTIAL ATTORNEY'S EYES ONLY

Table of Contents

CONTAINS INFORMATION DESIGNATED AS SAMSUNG HIGHLY CONFIDENTIAL ATTORNEY'S EYES ONLY

I. Introduction

1. I have been retained as an expert witness by Plaintiff and Counterclaim-Defendant Apple Inc. ("Apple"). I understand that the Defendants and Counterclaim-Plaintiffs in this case, Samsung Electronics Co., Ltd., Samsung Electronics America, Inc., and Samsung Telecommunications America, LLC (collectively, "Samsung"), have asserted U.S. Patent No. 7,698,711 to Moon-Sang Jeong ("the '711 patent") against Apple. I have been informed that Samsung is asserting claims 1-2, 7-10, and 15-18 of the '711 patent against several Apple products.

2. In this report, I provide my expert opinions regarding the validity of asserted claims 1-2, 7-10, and 15-18 of the '711 patent. I have been asked for my expert opinion as to whether the asserted claims of the '711 patent are valid or invalid. As discussed in more detail below, it is my opinion that the asserted claims of the '711 patent are invalid. In particular, the asserted claims of the '711 patent are anticipated and/or would have been obvious in light of the prior art, including commercial products, patents, and publications. Furthermore, the claims are invalid for lack of sufficient written description.

3. I previously submitted an expert report dated November 14, 2011 in this case regarding construction of the claim term "applet" in the '711 patent ("Claim Construction Expert Report"). I expect to testify at trial regarding the matters set forth in my expert reports, if asked about these matters by the Court or by the parties' attorneys.

II. Qualifications

4. I received a Bachelor of Science degree in Computer Science from the University of California, Riverside, in 1997. In 2001, I received my Ph.D. degree in Computer Science, also from the University of California, Riverside. My doctoral thesis, completed under the supervision of Professor Frank Vahid, was titled "Design Space Exploration of Parameterized System-on-a-Chip Architectures" and related to computer-aided design optimization of highly integrated circuits on chip.

5. I have been a member of the Department of Computer Science faculty at the University of California, Irvine ("UC-Irvine") since 2001. From 2001-2007, I held the title Assistant Professor of Computer Science. I was promoted to Associate Professor, with tenure, in 2007, and to full Professor in 2011. Beginning in 2011, in addition to my role as Professor of Computer Science, I was appointed Associate Dean for Student Affairs in the Donald Bren School of Information & Computer Sciences at UC-Irvine.

6. I have done extensive research in the area of embedded systems design. Embedded systems are devices that, in addition to having mechanical and electrical parts, make use of an embedded computing element, comprised of one or more processors and system software. A characteristic of embedded systems is the need for multitasking, i.e., the ability to perform a number of tasks simultaneously. My research is focused on software design for embedded systems, real-time operating systems, multitasking systems, cross-compilers, embedded processor architectures, multi-core processors, flash memory systems, low power design, and general system optimization algorithms. I have advised four Ph.D. students and am currently supervising and advising a group of five Ph.D. and two M.S. students. My research is supported by the National Science Foundation (NSF), with approximately $1.7M in current funding.

7. As a professor, I regularly teach both at the graduate and undergraduate levels. At UC-Irvine, among computer science and engineering students, I am probably best known for routinely teaching the upper division embedded systems course (CS 145). This course covers the design cycle of a typical embedded device, including all aspects of hardware and software integration. Additionally, I have taught courses in the areas of programming (including languages such as Java and C/C++), data structures and algorithms, logic design, modeling and simulation, ubiquitous computing, and compilers.

8. I have published over 70 peer-reviewed conference and journal papers, four of which have been recognized by Best Paper Awards. My papers are published in highly ranked and archived journals. I am a co-inventor on 10 issued US patents. I have co-authored two

popular textbooks on embedded system design that are widely used at top institutions in the US as well as around the globe. I received the prestigious 2011 Frederick Emmons Terman Award for my textbook entitled *Embedded System Design: A Unified Hardware/Software Introduction*.

9. Among my journal publications is a paper I co-authored with one of my graduate students, Andre Nacul, titled "Synthesis of Time-Constrained Multitasking Embedded Software," ACM Transactions on Design Automation of Electronic Systems, Oct. 2006, pp. 822-847, vol. 11 No. 4, ACM Press. This publication is cited on the face of the '711 patent under "Other Publications."

10. I am being compensated for my time working on the case at a rate of $275/hour.

11. I have attached as **Exhibit 1** my updated curriculum vitae setting forth my qualifications and publications.

III. Materials Reviewed

12. My opinions expressed in this report are based on my review of the following materials: the '711 patent, the prosecution history of the '711 patent and its parent application (U.S. Application No. 11/390,338), the prior art cited to the USPTO during the prosecution of the '711 patent, related prior art, Samsung documents, and any document cited in this expert report.

13. A full list of materials that I have reviewed in forming my opinions expressed in this report is attached as **Exhibit 2**.

14. My opinions also are based on my professional and academic experience including in the field of embedded systems and programming for multitasking operations.

IV. Understanding of the Law

15. I am not an attorney. For the purposes of this report, I have been informed about certain aspects of the law that are relevant to my analysis and opinions. My understanding of the law is as follows:

relevant industry or academic research experience in the areas of multitasking systems, embedded systems or programming for handheld devices. Alternatively, the ordinary artisan would have had a more advanced degree in computer science/engineering or a similar field with somewhat less additional work or research experience.

VI. Technology Tutorial

29. The following tutorial is a broad overview of a few subject areas in computer science that are relevant to the '711 patent claims and this report. For example, the '711 patent discloses a way of multitasking while playing a music file. The specific claims require certain types of computer programming structures, including one type called an "applet". Therefore, I summarize first how multitasking fits into the historical development of computing, explain what it is, and how it works. I then explain a certain class of computer languages called "interpreted" languages, and illustrate the point with the most common example, Java. Because understanding "applets" are helpful to understanding my opinions on the validity of the '711 patent, I explain them in the context of interpreted languages. Finally, because the '711 patent is first and foremost concerned with the play of digital music files called MP3 files, I explain the basics of MP3 technology as well.

A. Multitasking

30. Computers are typically comprised of numerous hardware components such as one or more central processing units (CPUs), main memory, storage devices, and peripheral devices (inputs/output). These hardware components are managed and integrated by a layer of software known as an *operating system*.

31. A *task* is typically computer programming that is intended to perform a specific function. In technical literature, the terms *process*, *thread*, or *task*, are often used interchangeably. An *application* (or an *app*) is composed of one or more tasks, and offers the user, for example, (i) a tool to perform an activity (e.g., word processing), (ii) a solution to solve a problem (e.g., calculator), or (iii) a service to network with others (e.g., web browser). An

application is *running* or *executing* if it is loaded into the computer's main memory and its instructions are being processed by the computer's CPU.

32. *Multitasking* involves the sharing of computer resources among a set of simultaneously executing tasks. Since tasks are the building blocks of applications, a multitasking system can simultaneously execute multiple applications. For example, a person, using her laptop computer, might simultaneously be writing a report using a word processing application and have a web browser application running on the side to receive Facebook status updates from friends and family.

33. In early days of computing, computers were expensive and their peripheral devices were slow. When an application reached a point that it needed input or output via a slow peripheral device, the CPU became inactive or "idle", which is an inefficient way for a computer to operate. To maximize CPU utilization, a form of multitasking known as *batch processing* was introduced. In batch processing, multiple applications were loaded into the main memory and made available to execute. When one application paused for input or output, another application was selected to execute on the CPU, keeping the CPU from idling. The earliest systems that performed this basic form of multitasking date back to the early 1960s. See, e.g., *Operating Systems and Concepts*, 4th Ed., at p.19.

34. Similarly, due to the expense and limited availability of early computers, the concept of *time-sharing* was introduced. Time-sharing allowed multiple users to run their applications on a single computer, hence spreading the cost among more users. The underlying technology that enabled time-sharing was multitasking, where the mix of applications came from different users rather than a single user. Time-sharing was the prominent model of computing during the 1960s and 1970s, representing at that time a major milestone in the history of computing. *Id.* A time-sharing system is by definition a *multi-user* system. In other words, a single system capable of accommodating multiple users simultaneously as if each user has her or his dedicated computer.

35. During the 1970s, a multitasking, multi-user, and time-sharing operating system by the name *UNIX* was being developed by a group of AT&T employees at Bell Labs. See "What is Unix?" published at www.unix.org ["Since it began to escape from AT&T's Bell Laboratories in the early 1970's, the success of the UNIX operating system has led to many different versions: recipients of the (at that time free) UNIX system code all began developing their own different versions in their own, different, ways for use and sale. Universities, research institutes, government bodies and computer companies all began using the powerful UNIX system to develop many of the technologies which today are part of a UNIX system."]. By the early 1980s, many universities (notably, UC Berkeley) had contributed to UNIX development, making it popular within universities. The popularity of UNIX within academia led to a number of early UNIX-based operating system products by startup companies (e.g., Solaris, Darwin, AIX, HP-UX and Linux). By the late 1990s, Apple introduced the Mac OS X, a UNIX-based operating system. By the mid to late 2000s UNIX-based operating systems were introduced for mobile devices.

36. The evolution of UNIX-based operating systems closely tracks the evolution of computers. Mainframe computers were expensive, could occupy an entire room, and were available only to large organizations such as government agencies. Minicomputers were physically scaled down versions of mainframes and made computing available to a broader class of corporate and government establishments. During the 1980-1990s, microcomputers became more widely available and bridged the eras of desktop and laptop computing. Mobile devices of the late 1990's to present time have placed computers in people's hands and pockets.

37. Operating systems with multitasking capabilities intended for mobile devices emerged by the late 1990's. For instance, the *Symbian* operating system, available since 1998, was specifically designed for smartphones by a group including Nokia and Ericsson. See Steven Vaughn-Nicholas, "OSs Battle in the Smartphone Market." *IEEE Computer* 36:10-12 (June 2003) at p.11 ("The Symbian OS was developed in 1998. Symbian Ltd. is owned by some of the world's biggest handset makers: Ericsson, Motorola, Nokia, Panasonic, Psion, Samsung

Electronics, Siemens, and Sony Ericsson."). For example, the Nokia 9210 Communicator, based on the Symbian operating system, was introduced in 2000 and available for sale by 2001. The 9210 shipped with applications such as calculator, clock, games, recorder, unit converter, text messaging, email, Internet browser, address book, calendar, and word-processing.

38. Parallel to the evolution of computers, another revolution in the form of computer networking -- the Internet -- was taking place. The early Internet was mainly used for exchange of static content (e.g., photos, documents, audio files, etc.). By 1995, the need to allow the exchange of dynamic content, for example an application, was answered by *Java*, a programming language made by Sun Microsystems. Java introduced the concept of applications intended to be exchanged over the Internet and executed anywhere and at anytime. Certain Java programs were referred to as *applets*.

39. Java allowed applications to be developed and distributed to multiple computers running different operating systems without re-writing the applications. Java became popular among Internet content providers as well as users. As with other innovations within the desktop computing area, mobile devices followed the trend and became Internet capable.

1. What is "Multitasking"?

40. As discussed above, multitasking involves the sharing of computer resources among multiple simultaneously executing tasks (*i.e.*, computer programs intended to perform well defined functions). A computer program can be understood as sequence of instructions that, when executed, for example, solves a problem. One can think of a computer program as a "cooking recipe." The CPU of a computer executes the program instructions. In my analogy, the CPU is the "cook". A task can be thought of as the process of executing the instructions of a computer program. In other words, a running task requires: (1) a computer program and (2) a CPU that is engaged in executing the program instructions. In my analogy, the task is the process of cooking, requiring both a cooking recipe and a cook.

41. It should be noted that, in a computer system, one typically finds multiple tasks, each associated with distinct computer programs. Conversely, one might discover multiple tasks, each associated with the same computer program.

42. In software systems, tasks are the smallest units of execution, and serve as building blocks of larger components, including applications. Today's applications are complex software systems integrating many functions and supporting numerous features. Today's personal computers are typically executing hundreds of tasks concurrently at any given time. Higher end computers, found in server houses, can potentially execute thousands of tasks concurrently at any given time. Mobile devices leading up to 2005 were typically executing far fewer tasks concurrently at any given time, perhaps tens instead of thousands. Thus servers, desktops and mobile devices all accommodate multitasking, but as computing resources become more limited as you go from servers to mobile devices, multitasking becomes more difficult as well.

43. A multitasking system is one that can accommodate multiple concurrent tasks. Multitasking requires the sharing of the computer resources, including the CPU, main memory, storage devices, and peripheral devices (inputs/output) among the many tasks. Sharing of computer resources is directed by the operating system. On a computer with a single CPU, the operating system might execute the instructions of each task for a short duration of time in a round robin fashion, in what is known as *task scheduling*. Task scheduling is usually driven by key policies such as fairness (i.e., each task receiving equal share of the CPU) and priorities (i.e., some tasks requiring urgent attention). The execution speed of the CPU as well as the rapid switching among tasks by the operating system far exceed the perception thresholds of the human senses, so it appears to a user that all tasks are executing in parallel.

2. Task Synchronization & Communication

44. Multitasking systems are typically designed to accommodate synchronization and communication among multiple running tasks. Task *synchronization* is the ability for task *A* to wait at a specific point within its instruction sequence for another task *B* to reach a specific point

within its instruction sequence. Task *communication* is the ability for tasks *A* and *B* to send and receive data to one another.

45. Task synchronization and communication are common elements of multitasking systems, from mainframe to mobile devices, as they provide the building blocks for the design of collaborative applications that provide a consistent and functional interface to the user.

46. In sum, a typical multitasking computer system will includes a number of applications and tasks synchronizing and communicating with each other in an orderly manner, as orchestrated by the operating system, to provide improved computer performance to the user.

B. Interpreted Languages & Host Applications

47. In some instances in computer programming, it is desirable for a first application to run within the context of a second application (i.e., the "host"), where the host application *interprets* and executes the instructions of the first application. This can be advantageous, for example, so that the host application can insulate or protect a user's computer from a potentially malicious first application that is obtained externally (e.g., from an unknown source on the Internet). Another advantage is that the first application can be accessible to a diverse user base and be implemented in essentially the same form across multiple different platforms.

48. That is, in software systems where a first application executes within the context of a second "host" application, the first application can be run independently of the platform on which the host application is executing. The platform encompasses aspects of a computer that include both hardware and the operating system. The host application provides the complete execution environment for the first application independently of the platform, including the operating system. For example, in 2005, a video game (i.e., the first application) could run on a web browser, such as Microsoft Internet Explorer (i.e., the host) running on a Windows-based PC (i.e., the platform). The same video game program could also run on a different host browser, such as Netscape on a laptop running the Linux operating system.

49. Applications executed within another host application operate using a restricted set of privileges, thus providing a high measure of security to the user. This is an important

property when applications are exchanged over the Internet between unknown entities. Without this security feature, content providers can create malicious applications that are unintentionally downloaded and executed by users, causing damage to the user's computer system (e.g., corruption of files or theft of personal data). Therefore, applications executing within another host application are allowed limited access to the user's operating system services, such as file system, network and main memory. The host application enforces this limitation and may provide to the user means to adjust the allowable level of privileges.

1. The Java Environment & Object Oriented Languages

50. *Java* is a popular computer programming language, made available by Sun Microsystems around 1995, and used to develop applications that run on a variety of platforms, including handheld devices, laptops, and desktops running different operating systems. A Java-enabled device is one that supports the necessary tools and environments needed to execute Java applications. Java or Java-like applications are developed once and distributed widely to users running different platforms, including different operating systems.

51. Another feature of Java is its object oriented programming paradigm. An *object oriented* programming paradigm is one where relevant data is contained in entities known as objects and a set of pertinent functions are associated with each object. For example, a programmer might create an object named "music" containing relevant data (a.k.a., properties) such as "title,", "album," "singer," and "duration." Moreover, the programmer might associate a set of functions with the "music" object such as "play," "pause," and "stop.

2. Applets

52. An *applet* is an application that is intended to run in the context of another application, namely the host application. Applets are often written in Java, so they are interpreted programs that inherit all the features and benefits of the Java framework.

53. In addition to being computer and operating system independent, a property inherit in all Java applications, applets are executed by the host application using a restricted set of privileges, thus providing a high measure of security to the user. Without this security feature,

content providers can create malicious applets that are unintentionally downloaded and executed by users, causing damage to the user's computer system (e.g., corruption of files or theft of personal data). Therefore, applets are allowed limited access to the user's operating system services, such as file system, network and main memory. The host application enforces these limitation on the applet, and may provide to the user means to adjust the allowable level of privileges. One example of a host application capable of executing applets is a web browser.

54. The underlying concept of applets as portable applications that are computer and operating system independent, and execute with limited privileges, is not limited to Java applets. Examples of other languages that support Java's interpreted model of execution are Flash, Javascript, Python, and PHP, to name a few.

C. Music Players and MP3 Format

55. Music is made of sounds and sounds are mechanical compressions or wave pressures. These waves are analog in nature, in other words they are continuously varying intensities over time. To capture music digitally, the analog waves are converted to electrical signals using, for example, a microphone. The electrical signal is in turn converted to digital values (i.e., a sequence of sound samples) by a component named an analog-to-digital converter (ADC). Once converted to the digital domain, the sequence of sound samples may be stored in a file, processed by the CPU, transmitted over the Internet, and so on. The same digital sequence of sound samples can be converted to analog form using a digital-to-analog converter (DAC). The output of the DAC can, for example, drive a speaker to recreate the original music.

56. A challenge in processing sound digitally is managing the sound samples. For high quality capture of music (e.g., compact disk (CD) quality), and for each audio channel, it is necessary to capture approximately 44,100 samples per second, each sample being two bytes long. A five minute song, thus, may require about 53 MB (million bytes) of storage (i.e., 2 ch x 44,100 samples/sec x 300 sec x 2 bytes/sample = 52,920,000 bytes). Therefore, 74 minutes of music requires about 780 MB of storage, which happens to be the typical storage capacity of

audio CDs. Management of audio data at this rate is impractical, in particular on mobile devices with limited network and storage capabilities.

57. MP3, which stands for the Moving Picture Experts Group (MPEG) level 3 audio encoding, offers a standard compression technique to aggressively reduce the size of audio files. MP3 can achieve 10 times reduction in size. A five minute song, thus, may require about five MB of storage. The MP3 standard governs both algorithms for compression as well as algorithms for decompression and has become a de facto standard in the industry.

58. A music player application can plays music, such as MP3 files. The core of a typical music player is a decompression task that is implemented according to the MP3 standard. Virtually any multitasking system can simultaneously play music while executing other applications. The term *background music player* is often used to highlight the fact that music playback requires little user interfacing (i.e., once a song or a playlist is chosen by the user, the user is likely to allow the music to play for extended periods of time with no further user-computer interactions) and hence once the playback task is initiated, it can be allowed to run with limited access to the display and input devices. A task with limited or no access to the display and inputs of a computer may be referred to as a *background task*.

D. Conclusion

59. The above tutorial is applicable to computers of varying vintages and complexity. From the early mainframe computers to mobile devices of early 2000s, one can readily establish consistent design principles that address the issues of multitasking from a software as well as a hardware perspective.

60. Multitasking techniques have been established for over 60 years, from at least the onset of UNIX, made widely available to industry and academia. *Operating System Concepts*, 4th Ed. (1994) at p.19. The '711 patent identifies an application of multitasking, as a way to eliminate the need for additional hardware resources (see, e.g., '711 patent at Col. 1:49-62) but offers no additional contribution to the technical field.

VIII. Claim Constructions Used in this Expert Report

93. The interpretation of the term "applet" in the asserted claims of the '711 patent is disputed between the parties. I understand that the parties have proposed claim constructions as set forth below.

Claim No.	Disputed Term	Apple's Proposed Construction	Samsung's Proposed Construction
1, 9, 17	"applet"	"An operating system-independent computer program that runs within an application module."	"A small application designed to run within another program."

94. I understand that as of the date of this report, no Claim Construction Order has yet been issued by the Court. My opinions set forth in this report are applicable under either party's proposed construction.

95. I understand that the parties have not proposed specific constructions, nor has the Court provided a construction, for any other term in the asserted claims of the '711 patent. As a result, I understand that I should give the claim terms the ordinary and plain meaning that a person of ordinary skill in the art would understand such terms to mean in the context of the '711 patent and its prosecution history. I have applied such plain meaning in my analysis.

IX. The Prior Art Renders the '711 Claim Obvious

A. Summary of Prior Art References

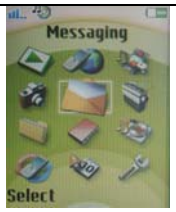
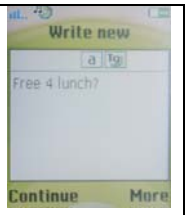
Sony K700i mobile phone and K700i reference guide

96. The Sony Ericsson K700i was publicly used in the US no later than March 21, 2004, as supported by the Sony Ericsson press release of that date indicating that the phone was on display at a trade show in Atlanta, GA (APLNDC-WH-A0000005351-5352). [redacted] [redacted] (APLNDC-WH-A0000026930-26932; APLNDC-WH-A0000024792-24793). Therefore, the device is prior art under 35 U.S.C. §102 because it was

publicly used more than one year before the '711 filing date and was on sale prior to any priority date to which Samsung is entitled. The accompanying K700i User Guide has a March 2004 publication date, more than a year before Samsung's claimed conception date of June 15, 2005 and priority filing date of August 30, 2005. The User Guide is therefore also prior art under 35 U.S.C. §102.

97. The K700i is a mobile phone with background music playback capabilities. The K700 was made available under the name K700i (*i* for international) in areas outside of Mainland China and K700c in Mainland China, featuring Chinese keypad and Chinese language interface. The K700i has built-in functions such as an email client, an HTML browser, a media player capable of playing MIDI, WAV, MP3 and AAC music files, and a 1.2 megapixel digital camera. The K700i incorporates the Java ME environment, including the Mobile Media API. As discussed below, the Java ME environment can serve as a platform for running applets. See, e.g., para. 177, 179, and 184.

98. The K700i is a multitasking phone with extensive background music playback capabilities one would expect from a music-enabled mobile phone. The K700i media player application can be selected from the main menu. Once selected, the media player application provides an interface to play a song from a list of songs or play a multi-song playlist. The media player application can be minimized to return to the standby mode: "Minimize – minimize the media player and return to standby. The music continues to play and you can use other applications and functions in the phone," K700i User's Manual at p.57. Once minimized, the K700i returns to the standby display, where alternate phone functions (e.g., phone book, calculator, calendar) may be used while music playback continues in the background. During music playback, including background music playback, a small music indicator is displayed at the top of the display. The background music playback capabilities of the K700i are illustrated in the following sequence of photos taken from an actual K700i mobile phone by me:

				
The media player application is selected.	A song by the name "Coo-coo" is played.	The music player application is minimized.	The messaging application is selected. Note the music icon on top left corner.	Music plays (see music icon) while messaging application is in use.

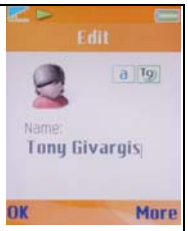
Sony Ericsson W800i and User Guide

99. The Sony Ericsson W800i User Guide (1st Ed.) was published in May 2005, before the August 30, 2005 filing of the priority application for the '711 patent priority date and describes the functions of the W800i phone. The W800i phone itself was released publicly on January 1, 2005 (APLNDC-WH0000015950), prior to the invention in the '711 patent and prior art under 35 U.S.C. §102. [redacted] (APLNDC-WH-A0000026930-26932; APLNDC-WH-A0000024792-24793).

100. The W800i is a mobile phone with background music playback capabilities. The W800i follows a similar design evolution as the Sony Ericsson K700i, having similar features and functionalities and adding elements such as a flight mode operation, 512 MB memory stick, and new packaging/cosmetic features. The W800i has Bluetooth, Infrared and USB connectivity as well as a 2 megapixel digital camera. The W800i is capable of playing MP3 and AAC music files and incorporates the Java ME environment, including the Mobile Media API.

101. The W800i is a multitasking phone with extensive background music playback capabilities. The W800i music player application is selected from the main menu. Once selected, the music player application provides an interface to play a song from a list of songs or

play a multi-song playlist. The music player application can be minimized to return to the standby mode: "Press [icon] to go to the WALKMAN player, or to minimize the WALKMAN player during playback," page 9, W800i User's Manual." Once minimized, the W800 returns to the standby display, where alternate phone functions (e.g., phone book, calculator, calendar) may be used while music playback continues in the background. During music playback, including background music playback, a small music indicator is displayed at the top of the display. The background music playback capabilities of the W800 are illustrated in the following sequence of photos taken from an actual W800i mobile phone by me.

				
The music player application is selected.	A song by the name "El Trumpet" is played.	The music player application is minimized.	The messaging application is selected. Note the music icon on top left corner.	Music plays (see music icon) while messaging application is in use.

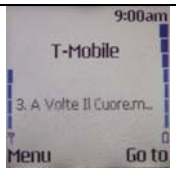
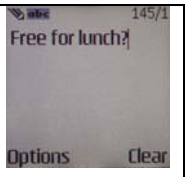
Nokia 3300 mobile phone and user guide

102. The Nokia 3300 User Guide was published in 2003 and the device itself was on sale in the U.S. no later than August 4, 2003 (APLNDC-WH-A0000005340), making both the user guide and the mobile phone it describes prior art under 35 U.S.C. §102 because they were publicly available more than one year before the filing date of the priority application leading to the '711 patent.

103. The Nokia 3300 is a multitasking phone capable of playing music in the MP3, AAC, Midi, and WAV formats. Moreover, the Nokia 3300 is able to play music in the background while the user engages other application such as, email, web browser, calendar and

text messaging in the foreground. As above for the Sony Ericsson K700i and W800i mobile phones, the Nokia 3300 incorporates the Java ME environment, including the Mobile Media API.

104. The Nokia 3300 music player application is selected from the main menu. Once selected, the music player application provides an interface to play a song from a list of songs or play a multi-song playlist. The music player application can be minimized to return to the standby mode. Once minimized, the Nokia 3300 returns to the standby display, where alternate phone functions (e.g., phone book, calculator, calendar) may be used while music playback continues in the background. When in standby mode, an indicator on the display indicates background music playback: "Standby Mode – Shows the track title when the Music player is on," Nokia 3300 User's Manual at page 22. The background music playback capabilities of the Nokia 3300 are illustrated in the following sequence of photos taken from an actual Nokia 3300 mobile phone by me.

				
The music player application is selected.	A song by the name "A Volte Il Cuore" is played. Selecting "Back" returns to standby.	The name of the song is indicated in the standby mode. Music continues playing.	The messaging application is selected. Music continues playing.	Music plays while messaging application is in use.

U.S. Pat. No. 7,123,945 to Kokubu

105. U.S. Pat. No. 7,123,945 , referred to as the Kokubu reference, first published as U.S. Application No. 2003/0119562 on June 26, 2003, more than a year before the earliest filing date of the '711 patent and therefore prior art under 35 U.S.C. §102.

106. Kokubo outlines methods to accommodate multitasking on a portable device, such as a mobile phone. In particular, Kokubu outlines methods to enable a user to perform a

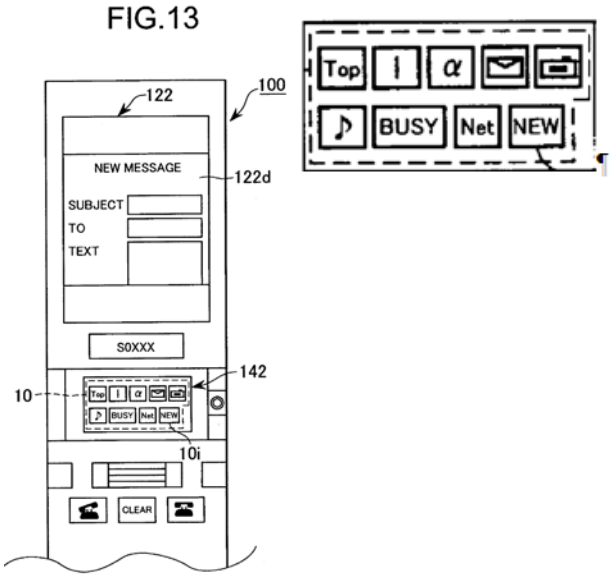
switching operation of the display among a set of applications that are executing in parallel, for example, browsing a web site while listening to music.

107. Of particular interest is the elaborate mechanisms outlined by Kokubu for generating icons to represent applications that are running in the background. The icons are displayed in one area of the screen while another area of the screen is used by a foreground application. The Kokubu patent establishes a method to switch among two or more applications that are executing in parallel, i.e., multitasking, where one of the applications is placed in one display region (e.g., the main screen area) while the remaining applications are represented as icons in a second display region (e.g., a smaller screen area on the top of the display).

108. Specifically, the Kokubo reference at Col. 13:4-9 teaches displaying an icon indicating music play ("For example, if, as shown in FIG. 10, the portable telephone 100 is used for listening to music, the audio player software is active, and the audio screen 122e is displayed on the main display unit 122, and a manually or automatically generated "♪" icon 10f is displayed on the sub-display unit 142.") Music play may continue even when other applications are in use, such as messaging or an incoming phone call. See, e.g., Kokubo at Col. 13:10-18, Col. 14:45-51, and FIGS 10-13.

109. For example, the Kokubo reference at FIG. 13 shows the display of the music play icon "♪" on the sub-display screen 142 when the user is composing a new message, as shown below (sub-display screen 142 of FIG. 13 is enlarged at right with music icon displayed in the lower left-hand corner):

FIG.13



110. As discussed above, during prosecution of the '711 patent, the PTO examiner rejected an earlier version of the asserted claims as obvious over Kokubo in view of the Senpuku application. On December 8, 2009, Samsung responded by amending the claims to include a limitation requiring "generating a music background play object, wherein the music background play object includes an application module including at least one applet." Samsung argued the amendment distinguished over Kokubo because "Kokubo makes no disclosure that the icon includes an application module, or that the application module includes at least one applet as instantly claimed." Amendment and Response of December 8, 2009 at p.10.

U.S. Pat. No. 6,928,648 to Wong

111. U.S. Pat. No. 6,928,648 to Wong et al. first published as U.S. Publication No. 2002/0156937 on October 24, 2002, more than a year before the earliest filing date of the '711 patent, making it prior art under 35 U.S.C. §102.

112. The Wong patent outlines an efficient mobile multimedia framework that can reside on a portable and mobile device with limited memory resources. The mobile multimedia framework provides objects and functionality to develop background music playing Java applications and applets, as highlighed in the following text from the patent: "[a]pplications use

these classes to create and control media playback. Additional implementation classes can be used to support the Application Programming Interface (API) by providing playback for Wave and MP3 audio files, QuickTime and MPEG-4 video files and Flash media formats," Wong patent at Col. 9:16-20, and "[t]hese programs include full-featured interactive, standalone applications, as well as smaller programs, known as applets, that run in a Java-enabled Web browser or applet viewer." Wong patent at Col. 1:31-34.

113. The Wong patent explicitly describes Java applets and describes the operating system-independent nature Java code. For example, Wong teaches that Java programs will run on any specific hardware and will run in any computer with the Java Virtual Machine software:

> Java is an interpreted language. The source code of a Java program is compiled into an intermediate language called "bytecode". The bytecode is then converted (interpreted) into machine code at runtime. Upon finding a Java applet, the Web browser invokes a Java interpreter (Java Virtual Machine), which translates the bytecode into machine code and runs it. Thus, Java programs are not dependent on any specific hardware and will run in any computer with the Java Virtual Machine software.

Wong at Col 5:5-13

114. Further, Wong specifically acknowledges the portability of Java that is a specific consequence of operating-system independence:

> The Java virtual machine layer 208 is developed to operate in conjunction with the native operating system 210 of the particular hardware 212 on which the mobile multimedia framework system 200 is to run. In this manner, Java applications 204 can be ported from one hardware device to another without requiring updating of the application code.

Wong at Col 4:41-50.

U.S. Pat. No. 6,526,041 to Shaffer

115. U.S. Pat. No. 6,526,041 to Shaffer et al. issued on February 25, 2003, more than one year before the earliest filing date of the '711 patent, and therefore is prior art under 35 U.S.C. §102.

116. U.S. Pat. No. 6,526,041, referred to as the Shaffer patent, outlines a method for music playback that involves the transfer of the music file and a media player applet to a client device. The specific example is that of a system where a server transmits the music file along with the media player applet to a client machine via a local area network (LAN). Alternatively, the music file is transmitted to the client machine via the LAN while the music player applet is pre-included on the client machine.

117. Shaffer discloses transferring an applet with a music file and music player to telephone callers while the callers are placed on hold, as described below:

> [A] first embodiment of the invention, when a caller is placed on hold, the music-on-hold system is configured to transfer a small application program or applet, having a music file and a media player, from the server to the client. Concurrently, a real time protocol (RTP) stream carrying music is used to provide music-on-hold to the client from the server during the applet transfer. This ensures that the held caller receives music at all times. The RTP stream is used in the above manner to provide music to held callers in all embodiments of the invention. In another embodiment of the invention, the applet includes only the media player which is transferred from the server to the client. The music provided by the RTP stream is saved by the client as an audio file on a mass storage device for later playback by the media player.

Shaffer patent at Col. 1:61-2:8.

U.S. Pub. App. 2005/0083642 to Senpuku

118. U.S. Pub. App. 2005/0083642, referred to as the Senpuku reference, published on April 21, 2005, prior to any claimed priority date for the '711 patent. Senpuku is therefore prior art under 35 U.S.C. §102 and was cited as prior art during prosecution of the '711 patent.

119. Senpuku describes a mobile phone having two display screens and a controller that displays content from a number of applications on the displays depending on the open/close state of one of the screens. Of particular interest is the methods outlined by Senpuku that provide mechanisms for switching an application from executing in the foreground to executing in the background and reverting the display to render a standby or default mode. Then, a method

is outlined to select another application from the standby mode to execute in the foreground. The combined effect is the ability to shuffle concurrently executing applications from foreground to the background, with a standby screen serving as a transition.

120. Senpuku also discloses a music player function. "In addition, the speaker 13 is not limited to reproducing voice signals but can also reproduce music and sound effects from electrical signals formed in a synthesisizer circuit, which is not shown in the figure." Senpuku at ¶ [0048].

U.S. Pub. App. 2003/0236814 to Miyasaka

121. U.S. Published Application 2003/0236814, referred to as Miyasaka, published on December 25, 2003, more than one year before the earliest filing date of the '711 patent and therefore prior art under 35 U.S.C. §102. Miyasaka was cited as prior art during the prosecution of the '711 patent.

122. The Miyasaka reference outlines a multitasking implementation, intended for resource constrained mobile devices. More specifically, Miyasaka describes a method to schedule a set of tasks (i.e., applications) on a processor where one of the tasks in particular is an audio playback (audio decoder) or recording (audio encoder) function. Miyasaka's contribution is in scheduling a high quality audio task at times when the load on the processor is light (e.g., when few tasks are running in parallel. Conversely, at times of heavy processor load (e.g., when many tasks are running in parallel), an audio task of lesser quality is scheduled in order to maintain the load on the processor at or below its capacity.

123. In the Accelerated Examination Support Document filed July 16, 2007, Samsung admitted that the Miyasaka application teaches at least the following elements of the '711 patent claims:

- A multi-tasking method in a pocket-sized mobile communication device, the method comprising:
- Selecting and playing a music file in the pocket-sized mobile communication device

- Displaying an indication that the music device is being played
- Selecting and performing at least one function of the pocket-sized mobile communication device while the playing of the music continues
- Continuing to display the indication that the music file is being played while performing the selected function
- Selecting a message function as required by claims 7 and 15
- A controller for selecting and playing a music file in the pocket-sized mobile communication device and selecting and performing at least one function of the pocket-sized mobile communication device while the playing of the music file continues as required by asserted independent claim 9
- A multi-tasking apparatus in a pocket-sized mobile communication device and a display unit for displaying an indication that the music file is being played as relates to independent claim 17

U.S. Pub. App. 2004/0077340 to Forsyth

124. U.S. Pub. App. 2004/0077340, referred to as the Forsyth reference, published on April 23, 2004 and is therefore prior art under 35 U.S.C. §102 because it published more than one year prior to the earliest filing date of the '711 patent.

125. Forsyth outlines a mobile device having a screen that displays realtime information of particular intrest to the user while the phone is not in use and essentially in idle mode. Forsyth specifically provides the example of a mobile device playing an MP3 music file and displaying information about the current track being played or the artist information on the standby screeen.

Q.H. Mahmoud, "The J2ME Mobile Media API"

126. The article by Q. H. Mahmoud titled "The J2ME Mobile Media API" has a publication date of June 2003, more than a year before the earliest filing date of the '711 patent, making it prior art under 35 U.S.C. §102.

127. The Mahmoud article describes the design of an MP3 player, with background playback capabilities intended for mobile phones. The specific example provided by Mahmoud

builds a "MIDlet" application. See, e.g., "J2ME Tutorial" provided at p.3 ("MIDlets are applets for mobile phones.")(APLNDC-WH-A0000025002). A MIDlet is an applet but with slightly restricted access to resources (such as display or memory). As with applets, MIDlets are usually written in Java, are object-oriented, are interpreted by a host application. On very resource-constrained systems, such a low-end mobile phones, a developer might chose to target a MIDlet rather than an applet, but the person of ordinary skill would recognize that the code segment disclosed in Mahmoud is consistent with what one would have to write in order to design an less-constrained applet for a mobile device.

128. Mobile phones leading up to 2005 commonly provided support for the Java Micro Edition (JME) and the Mobile Media API (MMAPI). The JME is a Java Virtual Machine (JVM) specification specifically designed for resource constrained mobile devices. Benefits of supporting the JME include an object oriented programming model and a device-independent API that facilitates rapid application design and deployment. The Mobile Media API (MMAPI) disclosed in Mahmoud is noted as being "designed to run with any protocol and format." Mahmoud at abstract. The Mahmoud article outlines an implementation of an MP3 player application with background play capabilities in as little as 29 lines of Java code. The MP3 play object, named "player" is constructed as follows:

```
Player player =
Manager.createPlayer(Manager.TONE_DEVICE_LOCATOR);
```

129. The Mahmoud article further demonstrates how an MP3 play object can be instructed to begin the music playback, for instance when a User Interface (UI) command is issued by the user, as follows.

```
// non-blocking start
p.start();
```

130. Of particular note is the "non-blocking" nature of the start method described in the paragraph above. A person of ordinary skill would recognize that a non-blocking method, in

Java, is a mechanism to achieve multi-tasking. In particular, it would be apparent to the ordinary artisan reading the Mahmoud article that an entity executing a non-blocking method is allowed to continue execution while a background entity assumes the function of the method.

131. The person of ordinary skill in the art would further recognize that Java applications and applets, and Java Micro Edition (JME), are operating-system independent. By 2005, Java programming was well-understood in the art, and, indeed, the attractiveness of Java was largely because of the portability of Java code from one system to another, as I describe in the Technology Tutorial above at para. 38-39.

B. The '711 Claims are Obvious In View of the Prior Art

132. In my opinion each of the asserted claims would have been rendered obvious by the prior art. As shown below, each of the elements of '711 patent claims 1, 2, 7-10, and 15-18 were taught by the prior art. A person skilled in the art would have been motivated to combine the prior art references for the reasons described below, and the combinations represent no more than known elements performing known functions to provide predictable results.

133. As discussed in the prosecution history summary provided above, the '711 claims were allowed over the prior art because Samsung amended the pending independent claims (claims 1, 9, and 17) to include a limitation requiring that the music background play object "includes an application module including at least one applet."

134. Samsung admitted during prosecution that at least asserted dependent claims 7, 8, 15, and 16 "have no features that would define over the references deemed most closely related if claims 1, 9, and 17 were found unpatentable." See para. 83 above.

135. It is my opinion that programming a mobile device with an application module including at least one applet for a music background play function would have been obvious to a person of ordinary skill in 2005, and would have been a straightforward and predictable modification of the prior art, to the extent the identified devices do not already use applets for the music play function.

136. In each of the proposed combinations of art below, the Wong patent, Shaffer patent, and Mahmoud article would, taken alone, have motivated the person of ordinary skill in the art to include an applet in a music background play application module. I discuss the reasons the ordinary artisan would have been motivated to include such an applet when programming the background music functions of the prior art.

137. For ease of reference, I have designated the elements of claims 1, 9, and 17 as elements [a]-[h] in the order they appear in the claim language.

1. The '711 claims are obvious over the Sony W800i alone or in view of the Wong or Shaffer patents or Mahmoud article

138. It is my opinion that Claims 1, 2, 7-10, and 15-18 of the '711 patent would have been obvious to a person of ordinary skill in the art in 2005 over the Sony W800i device and associated User Guide in view of Wong, Shaffer or Mahmoud.

139. Independent claim 1 would have been obvious to the person of ordinary skill in the art because all claim elements were present in the Sony W800i phone with the possible exception of the limitation requiring that the background play object includes an application module including at least one applet. While the presence of the J2ME environment in the phone would have encouraged the use of Java applets, the source code for the Sony W800i was not available for my inspection, and therefore I could neither verify nor rule out that it indeed contained an applet relating to music background play.

140. Attached as **Exhibit 3** is a claim chart that sets forth in detail where each element of claim 1 is found in the Sony W800i device and corresponding User Guide.

141. Claim 1 requires "a multi-tasking method in a pocket-sized mobile communication device including an MP3 playing capability." This element is disclosed by the Sony W800i phone and User Guide. See, for example, the Sony Ericsson W800i User Guide (1st Ed., May 2005) at pp.16-17: "Getting to know your phone" including "Internet Services", "Entertainment" including "MusicDJ", "File Manager", "Contacts", "Radio", "Camera", Messaging", "Walkman", and "Organizer". Furthermore, the User Guide describes the

"Walkman" player as "a music player and video player all in one." See Sony W800i User Guide at p.7.

142. An online review of the Sony W800i phone published online also indicates that this device has a music player with MP3 play functionality. See review of August 29, 2005 published online at http://infosyncworld.com/reviews/n/6112.html (APLNDC-WH0000006682-6684).

143. Finally, I have tested the Sony W800i device personally and confirmed that it is a pocket-sized mobile communication device that performs MP3 file playback capability. I have photographed relevant screenshots showing the device playing back music files and provided them above and in the chart at Exhibit 3.

144. **Element [a]** of claim 1 requires "generating a music background play object, wherein the music background play object includes an application module including at least one applet." The Sony W800i and associated User Guide disclose a mobile device that generates a music background play object as shown in the chart at Exhibit 3.

145. I personally played a music file on the Sony W800i by following the simple on-screen instructions. This included launching the music player from the menu, selecting a song, playing it, then minimizing the music player to return to the main menu. From there, I was able to select any of the remaining functions of the W800i while the music playback continued in the background. Throughout this process, a small icon on the top left gave in indication that music playback is in progress. In attached Exhibit 3 for claim 1, element [a], I have provided a screenshot, taken by me, illustrating the selection of the music player from the main menu by using the "Walkman" key. The Walkman key is described in the Sony Ericsson W800i User Guide at p. 15, while the guide at p.7 notes that the Walkman player works as a music player, providing instructions for playing music files at p.9.

146. Sony Ericsson W800i User Guide at p.34 notes the storage folder for Java applications and games while also providing an End-user license agreement for Java J2ME at p.2.

Taken together, the User Guide thus discloses that the Sony W800i is compatible with Java J2ME.

147. If the Sony W800i lacks a background music player application including at least one applet, which cannot be determined in the absence of the appropriate source code for the device, it nevertheless would have been obvious to the person of ordinary skill in the art in 2005 to include an applet in the application that controls music background play on the W800i.

148. Each of the Mahmoud article, Shaffer patent, and Wong patent disclose use of applets that would have motivated the ordinary artisan to use an applet to control a music player function.

149. For example, Wong teaches application modules including applets for music background play objects, specifically discussing the use of Java programming (see, e.g., Wong '648 patent at Col. 1:24-34), Java applets (e.g., Wong '648 patent at Col. 5:5-11), and discusses applications for providing playback for MP3 audio files (Wong at Col. 9:16-20). Given all the known advantages of programming using Java applets and Wong's disclosure of Java programming to support MP3 audio file playback, the Wong reference would have motivated the straightforward modification of the source code underlying Sony W800i music playback to include an applet.

150. The Mahmoud article would likewise have motivated the person of skill to include an applet for music player control functionality in the Sony W800i phone to the extent it lacks such programming. Mahmoud provides sample code for a music player applet compatible with the Mobile Media API (MMPAI) on J2ME-enabled devices (the Sony W800i is J2ME-enabled as discussed above), including the MP3 format. See Mahmoud at Abstract, p.1, and pp.8-10.

151. As discussed above, Shaffer discloses transferring an applet with a music file and music player to telephone callers while the callers are placed on hold. See para. 116-117.

> [A] first embodiment of the invention, when a caller is placed on hold, the music-on-hold system is configured to transfer a small application program or applet, having a music file and a media

> player, from the server to the client. Concurrently, a real time protocol (RTP) stream carrying music is used to provide music-on-hold to the client from the server during the applet transfer. This ensures that the held caller receives music at all times. The RTP stream is used in the above manner to provide music to held callers in all embodiments of the invention. In another embodiment of the invention, the applet includes only the media player which is transferred from the server to the client. The music provided by the RTP stream is saved by the client as an audio file on a mass storage device for later playback by the media player.

Shaffer patent at Col. 1:61-2:8.

152. **Element [b]** of Claim 1 requires "providing an interface for music play by the music background play object." The Sony W800i and associated User Guide each discloses this element. For example, the W800i User Guide illustrates the device and provides a key to the functions of the interface, which can be used to play music, including background music. See, e.g., Sony W800i User Guide at pp. 14-15. The Sony W800i device itself provides the interface that is illustrated in the cited portion of the User Guide.

153. **Element [c]** of claim 1 requires "selecting an MP3 mode in the pocket-sized mobile communication device using the interface." The Sony W800i phone and User Guide enables selection of an MP3 music player mode as explained in the associated User Guide at pp. 7-9. For example, selecting the "Walkman" icon on the standby screen opens the browser, allowing selection of the desired MP3 music file. The Walkman music player mode can also be directly accessed via the keys on the user interface by pressing the Walkman key on the keypad.

154. **Element [d]** of claim 1 requires "selecting and playing a music file in the pocket-sized mobile communication device in the MP3 mode." This element is taught by the Sony W800i device and User Guide, again as explained in the User guide at pp. 7-9. For example, after selecting the "Walkman" music play mode as described in the paragraph above, the user may browse for a desired song by title or alternatively by artist, highlighting the song and pressing the play button as shown on p.9.

155. **Element [e]** of claim 1 requires "switching the MP3 mode to a standby mode while the playing of the music file continues." The Sony W800i performs this function as

explained in the associated User Guide, which notes "[t]he standby screen is displayed when you are not calling or using the menus. This is where you can see, for example, the name of your service provider, the time and the date." Sony Ericsson W800i User Guide at p.18. I have personally tested the Sony W800i phone and confirm that music file play continues after switching to the standby screen. A screenshot, taken by me, is provided for illustration in attached Exhibit 3 at claim 1, element [e]. The small green arrow in the upper portion of the display indicates the music file play continues after switching into the standby screen, showing that the Sony W800i device also meets **element [f]** of claim 1, which requires "displaying an indication that the music file is being played in the standby mode."

156. **Element [g]** of claim 1 requires "selecting and performing at least one function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues." The Sony W800i device performs this function as described in the associated User Guide. See, e.g., User Guide at pp. 17 and 28, describing instructions for selecting the messaging function after bringing up the menu on the standby screen. After selecting "messaging" and "write new" options, the user may begin to compose a message. I have performed this function on the Sony W800i as directed in the User Guide and confirm that the user can perform this function while the playing of the music file continues, as indicated by the green arrow in the upper left of the display. See Exhibit 3 for screenshots, taken by me while performing the message function. I have further provided screenshots showing that music play function after selecting the calendar, stopwatch and alarm functions from the standby mode while the playing of a music file continues.

157. The continued presence of the music play indication, as indicated by the green arrow described above, further meets element [h] of claim 1, which requires "continuing to display the indication that the music file is being played while performing the selected function."

158. Dependent **claim 2** requires all the above limitations of claim 1, and further requires "wherein the displaying of the indication comprises displaying an icon." As described in the paragraph above, the Sony W800i continues to display the green arrow indication that the

music file is being played while performing a variety of functions selected from the standby mode. As shown in the screenshots I have provided in Exhibit 3, the green arrow display comprises an icon. Therefore, claim 2 is obvious over the same prior art as described above for claim 1.

159. Dependent **claim 7** of the '711 patent further requires "wherein the function selected from the standby mode comprises a message function." Because the W800i teaches the additional limitation of claim 7, claim 7 is obvious over the Sony W800i in view of Wong, Shaffer or Mahmoud for the same reasons discussed with respect to claim 1. The additional limitation requiring selecting a message function from the standby mode, which has been previously shown above at paragraph 156 with respect to claim 1, element [g].

160. **Claim 8** of the '711 patent requires all the elements of claim 1 and further requires "wherein the function selected from the standby mode comprises a phone-book function." Because the W800i includes the additional limitation of claim 8, claim 8 is obvious over the Sony W800i in view of Wong, Shaffer or Mahmoud for the same reasons discussed with respect to claim 1, and further as discussed below.

161. The Sony W800i and User Guide disclose selecting a phone book function from the standby mode. The Sony W800i User Guide explains explains "Phone Contacts: Contacts is like an address book, in which you can store names, phone numbers and email addresses. You can also add pictures, ringtones and personal information, such as birthday, email, Web and street addresses, to the contacts." Sony Ericsson W800i User Guide at p.24.

162. I have personally performed the selection of a phone book function from the standby mode by selecting the contacts icon from the standby menu option; I have provided a screenshot, taken myself and presented in Exhibit 3, showing music play continuing while this function is performed as evidenced by the green arrow icon in the upper left corner of the display.

163. As an alternative, instead of choosing the contacts display from the standby menu on the user display by selecting the appropriate icon, the User Guide explains that the user may push the keypad to bring up contacts. From the standby screen, the user presses and holds any of

keys 2-9 to reach a contact beginning with a specific letter on that key, as instructed by the User Guide at page 19.

164. Independent **claim 9** is very similar to claim 1, but recites an apparatus rather than a method. As shown in detail in the claim chart attached as Exhibit 3, the corresponding claim limitations may be found in each of the prior art references in the identical locations as provided above for claim 1. In my opinion, claim 9 is obvious over the W800i, alone or in view of Wong, Shaffer or Mahmoud for the same reasons discussed with respect to claim 1. The minor differences between claim 9 and claim 1 are discussed below.

165. In addition to the elements required by claim 1, claim 9 also requires a "controller" for generating the music background play object of element [a]. The '711 specification states that "[a] controller 110 controls the overall operation of the portable terminal." '711 patent at Col. 3:52-53. I have inspected the Sony W800i phone and verified that it comprises a processor, which the person of ordinary skill would recognize controls the overall operation of the portable terminal.

166. In addition to the elements required by claim 1, claim 9 also requires "a display unit for displaying an indication that the music file is being played in the standby mode." The Sony W800i phone and User Guide, as shown in the chart provided at Exhibit 3, disclose a display unit which indicates that a music file is being played in the background while in standby mode or while performing another function, such as messaging or opening the contacts phone book. The indication is presented on the display as a green icon in the shape of an arrow (see above for each of claims 1, 2, 7, and 8).

167. Dependent **claim 10** requires the same elements as claim 9, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claim 2 and is disclosed by the Sony W800i as discussed above. See para. 158 above. Therefore, claim 10 is obvious over the W800i in view of Wong, Shaffer or Mahmoud for the same reasons discussed with respect to claims 1-2 and 9.

168. Dependent **claim 15** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a message function." This additional limitation is the same as that required by claim 7 and is disclosed by the Sony W800i as discussed above for claim 7. See para. 159 above. Therefore, claim 15 is obvious over the W800i in view of Wong, Shaffer or Mahmoud for the same reasons discussed with respect to claims 1, 7 and 9.

169. Dependent **claim 16** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a phonebook function." This additional limitation is the same as that required by claim 8 and is disclosed by the Sony W800i as discussed above for claim 8. See para. 160 above. Therefore, claim 16 is obvious over the W800i in view of Wong, Shaffer or Mahmoud for the same reasons discussed with respect to claims 1, 8 and 9.

170. Independent **claim 17** is similar to claim 9, but does not require the "selecting and performing at least one function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues" limitation, nor does it require the "continuing to display the indication that the music file is being played while performing the selected function" limitation of claim 9. All elements of claim 17 are otherwise disclosed as described above for claim 9 and claim 1 in this section. Therefore, claim 17 is obvious over the W800i in view of Wong, Shaffer or Mahmoud for the same reasons discussed with respect to claims 1 and 9.

171. Dependent **claim 18** requires the limitations of claim 17 above, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claims 2 and 10 and is disclosed by the Sony W800i as discussed above for those claims. See para. 158 and 167 above. Therefore, claim 18 is obvious over the W800i, in view of Wong, Shaffer or Mahmoud for the same reasons discussed with respect to claims 1-2 and 9-10.

2. Motivation to combine the above references as claimed

172. I understand that the motivation to combine the teachings of the prior art references in the combinations identified above can be found in each of (1) the references themselves, (2) the nature of the problem being solved, (3) the express, implied and inherent teachings of the prior art, (4) the knowledge of persons of ordinary skill in the art, and (5) the predictable results obtained in combining the elements of the prior art.

173. As described above, the Sony W800i cell phone together with the accompanying Sony W800i User Guide disclose all elements of the asserted claims with the possible exception of a music background play object that includes an application module including at least one applet. I was not able to examine the source code to determine if the Sony W800i device included an applet in the application module responsible for background music.

174. One skilled in the art would have been motivated to modify the Sony W800i with the teachings of Wong, Shaffer or Mahmoud regarding applets because it would have been desirable for numerous reasons relating to the advantages of using applets in general, as described below, and because of specific disclosures in the references.

175. The nature of the problem being solved, as articulated in the '711 patent itself, was "a need for an improved system and method to allow a user to simultaneously work on multiple menus of the portable terminal while listening to music" without the additional cost and complexity of a dedicated control processor. '711 patent at Col. 1:49-51. The problem itself would have motivated the ordinary artisan in 2005 to look at applets which would obviate the need for additional hardware or software complexity, and preserve scarce computing resources.

176. One of ordinary skill in the art would have been motivated to employ mobile devices running Java or other applet-compatible applications, and in particular to use applets for the purpose of multitasking and playing of background music on a mobile device, in order to obtain (1) third party and post manufacturing addition of functionality, (2) added security against malicious software, (3) a method to accommodate multitasking, (4) forward

compatibility with features and updates developed at a point in future, and (5) portability of the code between operating systems.

177. Regarding (1) in the paragraph above, applets are designed against a well established specification, such J2ME mentioned earlier. Therefore, a third party software vendor could design and deliver functionality in the form of an applet to be installed on a Java enabled mobile device without proprietary engineering knowledge of the mobile device internals.

178. Regarding (2) above, applets execute within the context of another application. As such, the second application serves as a layer of security, limiting the level of access an applet may have to the underlying sub-systems, including data storage devices, where private user data may be stored.

179. Regarding (3) above, applet environments such as Java virtual machine provides mechanisms for multitasking, a requirement set forth in the J2ME standard for example. A mobile phone that adheres to the J2ME specification is, by definition, a multitasking phone, thus providing the well-known benefits of multi-tasking.

180. Regarding (4) above, a mobile enabled with a Java-like language may be released to the public by the manufacturer containing an applet with an unintentional software error (i.e., "bug"). In such circumstances, the manufacturer may make available a revised error-free applet that is downloaded to the mobile phone in order to correct the manufacturing software error. This scenario highlights how mobile phones enabled with Java-like languages are forward-compatible with software applications designed post manufacturing.

181. Regarding (5) above, applets can be implemented in essentially the same form across multiple different platforms. Because applets execute within the context of a second "host" application, the applet can be run independent of the platform on which the host application is executing. The host application provides the complete execution environment for the first application independently of the platform, including the operating system. The ordinary artisan would appreciate this advantage and write code that would not have to be re-written if needed for another operating system.

182. [redacted]

183. The Wong, Shaffer, and Mahmoud publications show that applets with music play function were well known in the art and it would have been obvious to the ordinary artisan that modifying the Sony W800i to include a music background play applet would be desirable. Each of these references discloses the use of applets specifically in the context of music player functions on phones, including mobile devices. Moreover, the presence of Java J2ME on the Sony W800i phone makes adding applet control even more straightforward because of the Java-applet compatible environment, further motivating the ordinary artisan to do so. The end-user license in the Sony W800i User Guide indicates that Java J2ME was used on the device (see Sony W800i User Guide at p.2).

184. Furthermore, implementing an applet for a music background play function on the Sony W800i mobile phone would have been a simple application of known technology for its known function to provide predictable results. The Sony W800i mobile phone is compatible with Java J2ME (see W800i User Guide at p.2), which means it supports running Java applications, which may include "applets" for performing specific tasks. The Wong, Shaffer, and Mahmoud articles provide evidence that music applets were well known prior to 2005, and the '711 patent claims require only the known use of an applet running within a music player application to provide an expected result, the ability to multitask while playing a music file in the background.

185. The Mahmoud article would have motivated the ordinary artisan to employ applets for running MP3 music files on mobile devices. As described in para. 126-131 above, Mahmoud describes implementation of an MP3 player application with background play capabilities in as little as 29 lines of Java code, a simple player that would be straightforwardly implemented as an applet running within a larger application. In 2005, a person of ordinary skill in the art would have appreciated the benefits of the J2ME environment as described in

Mahmoud , including a device-independent Application Programming Interface (API) that facilitated rapid application design and deployment. See Mahmoud at Abstract. Further, Mahmoud's teaching of a "non-blocking" start method, as explained above in para. 130, would have particularly motivated the ordinary artisan because such methods are particularly useful for multitasking functions.

186. Likewise, the Wong patent would have motivated the ordinary artisan to combine mobile devices with MP3 players including an applet because it discloses methods of running small media applications, including applets, independently of a device's native operating system. *See*, e.g., Wong patent at Col. 1:24-34 and Col. 9:16-20. Furthermore, Wong would have motivated the ordinary artisan to combine with the Sony W800i because it saves memory, important for mobile applications. See, e.g., Wong at Col. 2:45-47 ("Broadly speaking, the present invention fulfills these needs [for systems and methods providing mobile multimedia frameworks capable of operating on mobile computing devices] by providing a mobile multimedia framework, having a reduced memory footprint.")

187. The Shaffer patent would have motivated the ordinary artisan in 2005 to use an applet in a music background play object in any of the cited primary references because Shaffer teaches a system for providing music on a network by providing an applet having a music file and a media player from the server to the client. *See*, e.g., Shaffer at Col. 1:61-2:8. The Shaffer patent would also have motivated the person of ordinary skill to use an applet for music background play in the Sony W800i phone to the extent it lacks such programming, because using an applet for a music player function is described in Shaffer as conserving resources. Shaffer at Col. 2:49-51. The ordinary artisan would recognize that the resource-saving advantages of using an applet for music player function would be particularly important in resource-constrained devices such as mobile phones.

188. Furthermore, the results obtained by making any of the prior art combinations identified above would have been entirely predictable. The known elements of music player applets performing their known functions, playing MP3 music files, with the expected result of a

phone with multitasking capability when playing a music file would have been obvious whether combining the Sony W800i with Mahmoud, Shaffer, or Wong. Neither the specification of the '711 patent nor the associated file history indicates any unexpected results from the use of an applet within an application module in a music background play object. I cannot see any functionality in the '711 patent that would not have been expected from the combination of a well-known applet and a well-known music player on mobile device such as the Sony W800i.

189. In light of the above, one of ordinary skill in the art in 2005 would have found it obvious to combine the prior art teaching mobile devices with multitasking music functions, including displaying icons indicating background music play, with routine programming of well-known applets, such as Java 2 Micro Edition (J2ME) applications, including for MP3 player functions. It has been explained to me that according to the Supreme Court's standard in the *KSR* case, "[t]he combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results." *KSR Int'l Co.* v. *Teleflex, Inc*., 550 U.S. 398, 416 (2007). As described above, the asserted claims of the '711 patent represent the application of commonly known programming methods to existing mobile devices, with entirely predictable results.

3. The '711 claims are obvious over Kokubo and Senpuku in view of the Wong or Shaffer patents or Mahmoud article

190. It is my opinion that Claims 1, 2, 7-10, and 15-18 of the '711 patent would have been obvious to a person of ordinary skill in the art in 2005 over the Kokubo and Senpuku patents in view of either Wong, Shaffer or Mahmoud.

191. During the prosecution of the '711 patent as described above in Section VII C, the PTO examiner found that the combination of the Kokubo and Senpuku references rendered most of the pending claims obvious, including all the claims which eventually issued as the asserted '711 patent claims. See '711 Patent File History, Office Action of November 9, 2009 and discussion above at para. 87-88. Only after the applicant amended the claims to include the "applet" limitation were the claims allowed over this combination. It is my opinion that, had the

appropriate references been before the Examiner, it would have been clear that programming the background music play function application to include at least one applet would have been obvious. Each of the Mahmoud article, Shaffer patent, and Wong patent disclose use of applets as described above in Section IX A and each would individually motivate the ordinary artisan to use programming including an applet for providing music background play as otherwise provided by the combination of Kokubo and Senpuku. The reasons why the person of ordinary skill would have been motivated to combine the Kokubo and Senpuku references together along with any of the references motivating use of an applet are detailed below in Section IX B 4.

192. For clarity and brevity, I have not reproduced each of the referenced figures and illustrations from the prior art references discussed here; where necessary, I have cited to illustrations and longer quotations in the references and as provided in the chart attached as **Exhibit 4**.

193. Claim 1 requires "a multi-tasking method in a pocket-sized mobile communication device including an MP3 playing capability." I agree with the PTO examiner that Kokubo discloses a portable phone capable of multitasking, including music file play capability. For example, Kokubo that the device is operable for both "phone calls" and "music reproduction," and is further "capable of processing a plurality of application software (tasks) in parallel, and of generating icons for each of the application software (tasks) and switching between the tasks." See, e.g., Kokubo at Col. 10:52:62. See also, Kokobu at abstract, Col. 2:34-41, Col. 2:63-3:3, and FIGS. 10-13.

194. Senpuku likewise teaches "[a] multi-tasking method in a pocket-sized mobile communication device including an MP3 playing capability." The specification points out that the speaker for the disclosed portable phone device "is not limited to reproducing voice signals but can also reproduce music and sound effects" as discussed in Senpuku para. [0048]. Senpuku likewise teaches multitasking, including providing "a task display switching method" including "a portable apparatus and a portable communications apparatus which, when a plurality of

application software are activated and processed in parallel, make it possible to switch a display between each of the application software with ease.

195. **Element [a]** of claim 1 recites "generating a music background play object, wherein the music background play object includes an application module including at least one applet."

196. Kokubo teaches this element, as noted by the Examiner during prosecution. Kokubo discloses that "[f]or example, if, as shown in FIG. 10, the portable telephone 100 is used for listening to music, the audio player software is active, and the audio screen 122e is displayed on the main display unit 122, and a manually or automatically generated '♪' icon 10f is displayed on the sub-display unit 142." See also, Kokubo at Col. 13:4-10 ["In a portable apparatus capable of processing a plurality of tasks (application software) in parallel and of displaying a plurality of display regions which display data, an icon corresponding to a task (application software) displayed in a first display region serving as a main display region is generated automatically or manually, the generated icon is displayed in a second display region serving as a sub-display region."] *See also*, e.g., Kokubo at FIGS. 10-13, col. 2:63-3:3, 2:34-41, and abstract, provided in the attached chart.

197. **Element [b]** of claim 1 recites "providing an interface for music play by the music background play object." I agree with the PTO examiner that Kokobu teaches this element. Kokubo provides "an interface for music play by the music background play object" as shown, e.g., in Kokobu at FIG. 10, disclosing main display unit 122 with a plurality of sub-display regions for displaying icons including for music reproduction. See, e.g., Kokubo at Col. 2:34-41 and 12:44-48. Kokubo explains that "as shown in FIG. 10, the portable telephone 100 is used for listening to music, the audio player software is active, and the audio screen 122e is displayed on the main display unit 122, and a manually or automatically generated '♪' icon 10f is displayed on the sub-display unit 142." Kokubo at Col. 13:4-10. *See also*, e.g., Kokubo at FIGS. 1A, 1B, 3, 5, 7, and 11-13. Element [b] is therefore clearly met by Kokubo.

198. **Element [c]** of claim 1 recites "selecting an MP3 mode in the pocket-sized mobile communication device using the interface." I agree with the PTO examiner that Kokobu discloses this claim element. As discussed above for element [b], the portable phone disclosed in Kokobu is operable to process audio data, such as that in an MP3 music file. "In the portable telephone 100 shown in FIG. 10, an audio screen 122e is displayed on the main display unit 122, reproduction processing of audio data is carried out, and a plurality of the icons 10 are displayed on the sub-display unit 142." Kokubo at Col. 12:44-48. "For example, if, as shown in FIG. 10, the portable telephone 100 is used for listening to music, the audio player software is active, and the audio screen 122e is displayed on the main display unit 122, and a manually or automatically generated '♪' icon 10f is displayed on the sub-display unit 142." Kokubo at Col. 13:4-10

199. **Element [d]** of claim 1 recites "selecting and playing a music file in the pocket-sized mobile communication device in the MP3 mode." As described above for element [c], Kokobu teaches "reproduction processing of audio data" with an "audio screen" displayed on the main display unit. See Kokobu at 12:44-48. Kokubo also discloses that when the portable telephone is used for listening to music, the audio player software is active and the audio screen is displayed on the main display unit. Kokubo at Col. 13:4-10. From this disclosure, it is clear that the portable telephone of Kokubo is operable to select and play music in a music player mode.

200. **Element [e]** of claim 1 recites "switching the MP3 mode to a standby mode while the playing of the music file continues." I agree with the PTO examiner that the combination of Kokubo and Senpuku make this element obvious, because Kokobu, as disclosed above, teaches the playing of music files on the portable phone while processing other applications in parallel, i.e., multitasking. Senpuku teaches at ¶¶ 90, 105, 106, and 110.

201. Senpuku teaches at para. 90: "As explained in (1) to (3) above, the mobile communications device can execute a plurality of applications (for example, e-mail, function-setting, video replay, Internet browser). When the sub-display device 3b changes from closed status to open status and becomes usable, the execution result of the application currently being

executed on the main display device 3a is displayed on the sub-display device 3b. This operation is as explained in (1) to (3) above."

202. Further, Senpuku teaches "[0105] By the above processes, it becomes possible to always display the active screen, whether the sub-display device 3b is opened or closed. In addition, although the above explanation outlines a situation where one application is executed, it is also possible to execute a plurality of applications in parallel on respective display devices. When the sub-display device 3b changes to closed status, the handler (resource usage right) of the main display device 3a may be turned over to the application that was being executed on the active screen, and execution of the other applications which were not being executed on the active screen may be continued in the background. In addition, when continuing execution in the background, although the handler for the sub-display device 3b is transferred to other applications, screen updating may or may not be performed."

203. In addition, when executing a plurality of applications in parallel, the content displayed on the main display device 3a and the content displayed on the sub-display device 3b do not necessarily need to be related." Senpuku at para. [106]. Finally, at para. [110], " FIG. 18 explains how, when a standby screen is displayed, if the sub-display device 3b is pulled out, the sub-display device 3b operates applications pre-registered by the user. Here, an application for e-mail creation is displayed."

204. **Element [f]** of claim 1 recites "displaying an indication that the music file is being played in the standby mode." As discussed previously, Kokubo teaches that "as shown in FIG. 10, the portable telephone 100 is used for listening to music, the audio player software is active, and the audio screen 122e is displayed on the main display unit 122, and a manually or automatically generated "♪" icon 10f is displayed on the sub-display unit 142.

205. **Element [g]** of claim 1 recites "selecting and performing at least one function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues." Kokubo teaches multitasking in a mobile phone application as discussed above for elements [a]-[f]. See, e.g., Kokubo at Col. 9:57-Col.10:19 and Col. 10:52-62 ("The

portable telephone 100 is capable of processing a plurality of application software (tasks) in parallel, and of generating icons for each of the application software (tasks) and switching between the tasks.").

206. To the extent Kokubo is determined not to explicitly disclose a standby mode while performing multitasking operations, Senpuku discloses this element. Senpuku FIG. 18, as explained in the Senpuku specification at para.110, discloses "explains how, when a standby screen is displayed, if the sub-display device 3b is pulled out, the sub-display device 3b operates applications pre-registered by the user. Here, an application for e-mail creation is displayed." The standby screen in Senpuku would have been obvious to combine with the multitasking apparatus disclosed in Kokubu for the reasons discussed below.

207. **Element [h]** of claim 1 recites "continuing to display the indication that the music file is being played while performing the selected function." As discussed above, Kokubo teaches display of a music icon on a sub-display when the portable phone is being used for music play, and the music play continues even if another application (task) becomes necessary. For example, Kokubo teaches:

> For example, if, as shown in FIG. 10, the portable telephone 100 is used for listening to music, the audio player software is active, and the audio screen 122e is displayed on the main display unit 122, and a manually or automatically generated "♪" icon 10f is displayed on the sub-display unit 142. If under such circumstances, a telephone call is received, it is automatically switched from the audio player software to a telephone call software (or by operating the input/call keys 162). As shown in FIG. 11, the main display unit 122 switches to a call data screen 122b which displays the telephone number of the person on the line, the duration of the call and the like. The audio player software in the main region in the memory is shifted to the sub-region, and reproduction of music is **continued** or temporally stopped.

Kokubo at Kokubo at Col. 13:4-21 (bold emphasis added).

See also, e.g., at Col. 14:16-30 and FIG. 10.

208. Dependent **claim 2** requires all the above limitations of claim 1, and further requires "wherein the displaying of the indication comprises displaying an icon." As shown in

the chart attached as Exhibit 4 and discussed extensively in para. 204 and 207 above as relating to the "indication" element of claim 1, the Kokubo patent discloses a portable phone generated '♪' icon which is displayed while background music file is being played. Kokubo at Col. 13:4-10, Col. 14:17-31, and FIG. 10.

209. Dependent **claim 7** of the '711 patent depends on claim 1 and further requires "wherein the function selected from the standby mode comprises a message function." The combination of Kokubo and Senpuku patents disclose this element as provided below. Furthermore, I agree with the original PTO Examiner that with regards to claim 7, Kokubo teaches switching between selected functions, including messaging functions and telephone functions. Kokubo at col. 10:54-62. Switching between applications/functions is disclosed in Kokubo, as is email messaging (col. 9, line 56-col. 10, line 21; col. 10, lines 59-62). With regards to **claim 8** (adding the limitation of a phone book function being selected from a standby screen), the limitation would have been obvious to the person of ordinary skill that using the calling function on a portable phone would involve a phone book for stored contacts. The PTO examiner found as follows in rejecting claims 7 and 8 and obvious over the combination of Kokubo and Senpuku:

> As to claims 7 and 8, KOKUBO teaches the switching execution between selected functions wherein the functions are a telephone function or a message function (via the portable telephone is equipped with multiple functions including telephone calling functions and creation, transmissions, and reception of e-mail functions)(col.10, lines 54-62)(and switches between the software performing the different tasks. It is obvious to one of ordinary skill in the art that a telephone calling function invokes the phone book of the telephone to call a registered number. SENPUKU teaches that the switching of tasks on a mobile telephone involves moving the current function to the sub-display and displaying the standby screen. It is well known in the art that the stand-by screen allows for users to invoke functions for execution. Therefore, the combination with the support of the well known teachings discloses the limitations as detailed, e.g., invoking phonebook/messaging functions via a standby screen.

Office Action of November 9, 2009 at p.7.

210. I agree with the PTO examiner's analysis above. Initiating mobile device functions from a standby screen menu was not novel in 2005, evidenced by at least this combination of art but from numerous other devices as detailed elsewhere in this report. To the extent Kokubo does not explicitly disclose choosing a messaging function or a phone book function from a standby screen, Senpuku discloses this element and would have been obvious to combine with the teachings of Kokubo for reasons discussed below.

211. With regard to **claim 7**, Senpuku teaches "FIG. 18 explains how, when a standby screen is displayed, if the sub-display device 3b is pulled out, the sub-display device 3b operates applications pre-registered by the user. Here, an application for e-mail creation is displayed." Senpuku at ¶ 110; *see also* at FIG. 18. Thus, the function executed from a standby screen includes at least a message function as provided by the "application for e-mail creation" in the excerpt above.

212. **Claim 8**, as described above, requires the same limitations as claim 1 but additionally recites "wherein the function selected from the standby mode comprises a phone-book function." Choosing functions selected from the standby mode is disclosed by Senpuku as explained above for claim 7, while Kokobu discloses making phone calls where the application runs in parallel with other functions as described for claim 7 and disclosed, e.g., at Kokubo at Col. 10:52-62, Col. 9:56-Col.10:19, and Col. 13:42-47. I agree with the examiner that "[i]t is obvious to one of ordinary skill in the art that a telephone calling function invokes the phone book of the telephone to call a registered number." For these reasons, Kokubo and Senpuku in combination make the invention in claim 8 obvious.

213. Independent **claim 9** is nearly identical to claim 1 and the claim limitations may be found in the identical locations as provided above for claim 1.

214. In addition to the elements required by claim 1, claim 9 also requires a "controller" for generating the music background play object of element [a]. The '711 specification states that "[a] controller 110 controls the overall operation of the portable

terminal." '711 patent at Col. 3:52-53. Kokubo discloses a "control unit 115" as described in detail at Col. 6:33-64 that meets this limitation.

215. Claim 9 also requires "a display unit for displaying an indication that the music file is being played in the standby mode." Kokubo discloses a display unit for displaying an indication that the music file is being played in the background while multitasking. See Kokubo, e.g., at FIG. 14, disclosing a main display unit 122 subdivided into multiple sub-display regions. See also, Kokubo at Col. 14:52-58 and FIGS. 1a, 3, 5, 7, and 10-13. For the reasons above, claim 9 is obvious over the same combination of Kokubo and Senpuku as described for claim 1.

216. Dependent **claim 10** requires the same elements as claim 9, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claim 2 and is disclosed by Kokubu as discussed above. See para. 208.

217. Dependent **claim 15** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a message function." This additional limitation is the same as that required by claim 7 and is disclosed by Kokubu and Senpuku as discussed above for claim 7. See para. 209-211.

218. Dependent **claim 16** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a phonebook function." This additional limitation is the same as that required by claim 8 and is disclosed by Kokubu and Senpuku as discussed above for claim 8. See para. 209, 210, and 212.

219. Independent **claim 17** requires similar limitations as claim 9, but does not require the "selecting and performing at least on function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues" limitation, nor does it require the "continuing to display the indication that the music file is being played while performing the selected function" limitation of claim 9. All elements of claim 17 are otherwise disclosed as described above for claim 9 and claim 1 discussed above.

220. Dependent **claim 18** requires the limitations of claim 17 above, and further requires "wherein the indication comprises an icon." This additional limitation is the same as

that required by claims 2 and 10 and is disclosed by Kokubu as discussed above for those claims and not repeated here.

4. Motivation to combine the above references as claimed

221. The person of ordinary skill would have been motivated to combine the teachings of Kokubo with the standby mode taught by Senpuku. As discussed above, when the sub-display in Senpuku is closed, the active screen on the display continues to execute the application other applications are continued in the background. *See, e.g.*, Senpuku publication at paragraphs ¶¶ 105, 106, 110. The ordinary artisan would therefore recognize that the combination of Senpuku and Kokubo would the user to switch into a standby mode with an MP3 file still continuing to play. Further, the music play indication icon taught in Kokubo would have been obvious to combine with the standby screen disclosed in Senpuku. The person of ordinary skill would have found it desirable to indicate music file play on the display in standby mode, and the convenience of selecting and performing a function of the phone from the standby screen while the music play continued.

222. As described above in paragraphs 175-189 describing the known advantages of applets, the ordinary artisan would have read the Wong, Shaffer, or Mahmoud references and been motivated to add music player functionality to the Kokubo and Senpuku device. Any programming environment, whether Java or otherwise, compatible with using applets would have provided the well-appreciated advantages of applets for mobile devices, particularly as they relate to multi-tasking. The specific teachings of Wong, Shaffer, or Mahmoud also provide added motivation to use applets for music play function in order to preserve computing resources. Rather than repeat the details of these teachings, please refer back to paragraphs 175-189, which are applicable for the combination of Kokubo and Senpuku as well as for the Sony W800i and accompanying User Guide because applets would confer the same advantages to a mobile device in either scenario.

5. The '711 claims are obvious over the Sony K700i in view of the Wong or Shaffer patents or Mahmoud article

223. It is my opinion that Claims 1, 2, 7-10, and 15-18 of the '711 patent would have been obvious to a person of ordinary skill in the art in 2005 because (1) all elements of claim 1 were present in the prior art, including in the Sony K700i device and associated User Guide in combination with any of the Wong patent, Shaffer patent, or Mahmoud article.

224. Independent claim 1 would have been obvious to the person of ordinary skill in the art because all claim elements were present in the Sony K700i phone with the possible exception of the limitation requiring that the background play object includes an application module including at least one applet. However, it would have been obvious to the person of ordinary skill in the art in 2005 that including an applet in an application module controlling music background play function would be desirable for numerous reasons as discussed below in para. 253.

225. **Claim 1** requires "a multi-tasking method in a pocket-sized mobile communication device including an MP3 playing capability." The Sony Ericsson K700i mobile phone and User Guide teach this limitation. See, e.g., Sony Ericsson K700i User Guide (1st Ed., March 2004) at p.9: "Getting to know your phone" including "Internet Services", "Entertainment" including "MusicDJ", "Camera", "File Manager", "Phonebook Contacts", "Radio", Messaging", "Media Player", "Organizer", and "Display". See also K700i User Guide at p.63. ["The media player works as a music player and video player all in one. You can play music and video clips that you receive in a picture message, or that you download to your phone. The following are supported by the media player: MP3, MPG4, 3GPP and WAV (with 16 kHz as maximum sample rate)."]

226. Finally, I have tested the Sony K700i device personally and confirmed that it is a pocket-sized mobile communication device that performs MP3 file playback capability. I have photographed relevant screenshots showing the device playing back music files and provided them above and in the chart at **Exhibit 5**.

227. **Element [a]** of claim 1 requires "generating a music background play object, wherein the music background play object includes an application module including at least one applet." The

Sony K700i and associated User Guide disclose a mobile device that generates a music background play object as shown in the chart at Exhibit 5. See, e.g., K700i User Guide at p.63 ("You can listen to music and view video clips that you have saved in your phone. The music list keeps on playing until you press Stop."). The User Guide also describes, at p.98, the object on the display screen that selects the audio/video player application. Please see the screenshot, taken by me, as provided in the chart at Exhibit 5.

228. I played a music file on the K700i phone with ease by following the simple on-screen instructions. This included launching the music player from the menu, selecting a song, playing it, then minimizing the music player to return to the main menu. From there, I was able to select any of the remaining functions of the K700i while the music playback continued in the background. Throughout this process, a small icon on the top left gave in indication that music playback is in progress. The screenshot provided in the chart, taken by me, illustrates the selection of the music player from the main menu.

229. I note that the K700i User Guide "icon description" at p.99 also describes an icon that indicates a Java application is running, indicating the K700i phone would be expected to be compatible with Java applets for the music player function.

230. If the Sony K700i lacks a background music player application including at least one applet, which cannot be determined in the absence of the appropriate source code for the device, it nevertheless would have been obvious to the person of ordinary skill in the art in 2005 to include an applet in the application that controls music background play on the K700i.

231. Each of the Mahmoud article, Shaffer patent, and Wong patent disclose use of applets that would have motivated the ordinary artisan to use an applet to control a music player function in combination with the Sony K700i and associated User Guide for the same reasons as given above for the Sony W800i phone and User Guide. See explanation above at para. 172-189.

232. **Element [b]** of Claim 1 requires "providing an interface for music play by the music background play object." The Sony K700i and associated User Guide each discloses this element. For example, the K700i User Guide illustrates the device and provides a key to the functions of the interface, which can be used to play music, including background music. See,

e.g., Sony Ericsson K700i Manual at p.8. The Sony K700i device itself provides the interface that is illustrated in the cited portion of the User Guide and provided by me in the screenshot in Exhibit 5.

233. **Element [c]** of claim 1 requires "selecting an MP3 mode in the pocket-sized mobile communication device using the interface." The Sony K700i phone and User Guide enables selection of an MP3 music player mode as explained in the associated User Guide. See, e.g., User Guide at pp. 63, directing the user to select the Media Player icon from the menu screen and select a file before pressing the Play button.

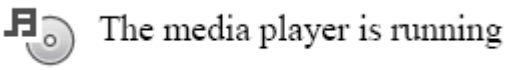
234. **Element [d]** of claim 1 requires "selecting and playing a music file in the pocket-sized mobile communication device in the MP3 mode." This element is taught by the Sony K700i device and User Guide, again as explained in the User guide at pp. 63.

235. I followed the instructions, selected, and played a music file on the K700i phone. The screenshot provided in Exhibit 5, taken by me, illustrates the music player application half way through the playback of a song by the name Ritmo Ratmo.

236. **Element [e]** of claim 1 requires "switching the MP3 mode to a standby mode while the playing of the music file continues." The Sony Ericsson K700i enables "switching the MP3 mode to a standby mode while the playing of the music file continues." For example, see Sony Ericsson K700i User Guide at 63, which describes selecting the "minimize" option to switch to the standby mode ("Minimize—minimize the media player and return to standby. The music continues to play and you can use other applications and functions in the phone"). I operated the K700i phone and switched into a standby mode while the music play continued.

237. The two screenshots for this element, taken by me and provided in the attached chart, illustrate the mechanism of the K700i of reverting back to the standby screen while the music continues to play in the background. The first image shows the menu that is loaded when selects the "More" option of the music player. The music player provides a number of features, one of these features is the minimize command to send the music player to the background. Once minimize is selected, the K700i reverts back to the standby mode. One is free to execute another function of the mobile phone from the standby mode as music continues.

238. The small music play icon resembling a musical note superimposed on a music record in the upper portion of the display indicates the music file play continues after switching into the standby screen, showing that the Sony K700i device also meets **element [f]** of claim 1, which requires "displaying an indication that the music file is being played in the standby mode." Again, I have provided a screenshot for element [f], taken by me, in the attached chart. The icon is described in the User Guide at pp.98-99 and is reproduced below as shown in the User Guide:



239. **Element [g]** of claim 1 requires "selecting and performing at least one function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues." The Sony K700i device performs this function because multiple functions may be selected from the standby mode including messaging, stopwatch, and calendar functions.

240. I have provided two images in the associated chart, taken by me, to illustrate how the menu of the K700i is selected in order to start the text messaging feature. Once text messaging was selected, I was able to complete a text message while music was playing in the background. Note that the icon indicator continues to display. Please see the additional images in the chart for element [g], taken by me, to illustrate the operation of the stop-watch and calendar applications while music playback continues. Note that the icon indicator continues to display in each case, thus meeting **element [h]** of claim 1, which requires "continuing to display the indication that the music file is being played while performing the selected function."

241. Dependent **claim 2** requires all the above limitations of claim 1, and further requires "wherein the displaying of the indication comprises displaying an icon." In the chart provided for claim 2, as for claim 1 above, I provide screenshots I took of the Sony K700i which demonstrate that an icon (appearing as a musical note superimposed on a record or music disc) is displayed while background music file is being played.

242. Dependent **claim 7** of the '711 patent requires all elements of claim 1 and further requires "wherein the function selected from the standby mode comprises a message function." The Sony K700i device and User Guide disclose this element as shown above for claim 1 element [g], in which I selected a messaging function from standby. See K700i User Guide at p.41 for

instructions on using the messaging feature of the K700i phone. See chart at Exhibit 5 for screenshots I have provided showing the messaging function being selected.

243. Dependent **claim 8** of the '711 patent requires all the elements of claim 1 and further requires "wherein the function selected from the standby mode comprises a phone-book function." The Sony K700i and User Guide disclose selecting a phone book function from the standby mode. See, e.g, User Guide at p.24.

244. I have also provided screenshots in Exhibit 5 showing the selection of the phonebook function from a standby menu. The screenshots, taken by me, illustrate how one can reach out to view contact information stored within the phone while music playback continues in the background. Note the indicator icon on the top left corner of the display as music play continues.

245. Independent **claim 9** is nearly identical to claim 1 and the claim limitations may be found in the identical locations as provided above for claim 1.

246. In addition to the elements required by claim 1, claim 9 also requires a "controller" for generating the music background play object of element [a]. The '711 specification states that "[a] controller 110 controls the overall operation of the portable terminal." '711 patent at Col. 3:52-53. I have inspected the Sony K700i phone and verified that it comprises a processor, which the person of ordinary skill would recognize controls the overall operation of the portable terminal.

247. Claim 9 also requires "a display unit for displaying an indication that the music file is being played in the standby mode." The Sony K700i and User Guide disclose a display unit, as shown above for the user interface element in claim 1, element [b]. A screenshot of the display unit is provided in the attached chart, while the K700i User Guide provides an illustration of the display at p.8.

248. Dependent **claim 10** requires the same elements as claim 9, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claim 2 and is disclosed by the Sony K700i as discussed above. See paragraph 241.

249. Dependent **claim 15** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a message function." This additional limitation is the same as that required by claim 7 and is disclosed by the Sony K700i as discussed above for claim 7. See paragraph 242.

250. Dependent **claim 16** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a phonebook function." This additional limitation is the same as that required by claim 8 and is disclosed by the Sony K700i as discussed above for claim 8. See paragraph 243-244.

251. Independent **claim 17** requires similar limitations as claim 9, but does not require the "selecting and performing at least on function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues" limitation, nor does it require the "continuing to display the indication that the music file is being played while performing the selected function" limitation of claim 9. All elements of claim 17 are otherwise disclosed as described above for claim 9 and claim 1.

252. Dependent **claim 18** requires the limitations of claim 17 above, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claims 2 and 10 and is disclosed by the Sony K700i as discussed above for those claims.

6. Motivation to combine the above references as claimed

253. The ordinary artisan in 2005 was fully aware of the advantages of using applets and even in the absence of explicit teachings in the references would have been motivated by the known advantages to use an applet for the music background object as claimed in the '711 patent. Please see above at para. 172-189 for an explanation of the known advantages of programming applets in Java or similar languages. The person of ordinary skill in 2005 would have further read the Wong, Shaffer, or Mahmoud references and been motivated to add music player functionality to the Sony K700i device as disclosed by the phone and accompanying User Guide. *Id*.. The Sony K700i, like the Sony W800i, is also a Java J2ME compatible phone and in fact

has a specific icon for display when the phone is running a Java-based application. See Sony K700i User Guide at p.99 and end-user license agreement at p.2. Sony Ericsson K700i User Guide at p.34 notes the storage folder for Java applications and games. Taken together, the User Guide thus discloses that the Sony K700i is compatible with Java J2ME, which the person of ordinary skill would understand is an ideal environment for applet control of music player function.

7. The '711 claims are obvious over the Nokia 3300 and Miyasaka or Kokubu references in view of the Wong patent, Shaffer patent, or Mahmoud article

254. It is my opinion that Claims 1, 2, 7-10, and 15-18 of the '711 patent would have been obvious to a person of ordinary skill in the art in 2005 because (1) all elements of claim 1 were present in the Nokia 3300 device and associated User Guide, together with either of the Miyasaka or Kokubu references in view of any of the Wong, Shaffer, or Mahmoud references and (2) the ordinary artisan would have been motivated to combine the teachings of the prior art in the manner claimed in the '711 patent and doing so would require only the simple application of known elements to perform known functions with predictable results. Thus, the difference, if any, between what is taught by the prior art and what is claimed in the '711 patent would have been obvious to the person of ordinary skill in the art by 2005.

255. Independent **claim 1** would have been obvious to the person of ordinary skill in the art because all claim elements were present in the Nokia 3300 phone and associated User Guide with the possible exceptions of the limitation requiring that the background play object includes an application module including at least one applet, and the limitation requiring display of an indication, such as an icon, when a music file is being played in the standby mode. However, it would have been obvious to the person of ordinary skill in the art in 2005 that including an applet in an application module controlling music background play function would be desirable for numerous reasons as discussed below in paragraphs 175-189.

256. Independent **claim 1** requires "A multi-tasking method in a pocket-sized mobile communication device including an MP3 playing capability." The Nokia 3300 mobile phone and accompanying Extended User Guide each teaches a device with MP3 file playing capability that is capable of multitasking. "You can listen to MP3 and AAC music files stored on the memory card in your phone with the Music player, or you can listen to the Radio." Nokia 3300 and Extended User's Guide at p.38. See attached chart in **Exhibit 6**.

257. Miyasaka also teaches a mobile device with a multitasking ability, including an MP3 playing capability. See, e.g., Miyasaka at para. [0020]. During prosecution, Samsung admitted Miyasaka teaches this element of the claim. See discussion of prosecution history above.

258. **Element [a]** of claim 1 recites "generating a music background play object, wherein the music background play object includes an application module including at least one applet." The Nokia 3300 and Extended User Guide teach generating music background play, as I tested myself.

259. I played a music file on the Nokia 3300 with ease by following the simple on-screen instructions. This included launching the music player from the menu, selecting a song, playing it, then minimizing the music player to return to the main menu. From there, I was able to select any of the remaining functions of the Nokia 3300 while the music playback continued in the background. The image presented in the accompanying chart in Exhibit 6, taken by me, illustrates the selection of the music player from the main menu.

260. I also note that the Extended User Guide points out that the Nokia 3300 "supports Java 2 Micro Edition (J2ME), which is a version of Java technology specifically designed for small consumer products. The phone includes some Java applications and games, and supports downloading new applications and games from different WAP services." Nokia 3300 Extended User's Guide at p.18. As discussed above, J2ME provided the perfect environment to execute Java applets for a music player functionality. A person of skill in the art would have been motivated to modify the Nokia 3300 to include an applet because of the known advantages of

applets, particularly when the phone is J2ME-compatible as is the Nokia 3300 phone. For the same reasons as described above for the Sony W800i and Sony K700i, the person of ordinary skill would have been motivated by the teachings of Wong, Shaffer, or Mahmoud to include a music player function including at least one applet. See para. 172-189 specifying relevant disclosure in these three references above.

261. Miyasaka also generates a music background play object because it plays digital music files and displays an indication that music play is occurring during multitasking. See Miyasaka FIGS. 17A-D.

262. **Element [b]** of claim 1 requires "providing an interface for music play by the music background play object." The Nokia 3300 mobile phone and Extended User Guide both teach the user interface for music play by the music background play object. For example, see the Extended User Guide at p.20, and further the photograph, taken by me, of the Nokia 3300 user interface. The photograph is presented in the corresponding chart in Exhibit 6.

263. Miyasaka also teaches an interface for music play. For example, see illustrated user interface for the mobile phone disclosed in Miyasaka at FIG. 11 (provided in attached chart for Exhibit 6). See also, Miyasaka at para. [0012-0013].

264. **Element [c]** of claim 1 requires "selecting an MP3 mode in the pocket-sized mobile communication device using the interface." The Nokia 3300 and Extended User's Guide teach "selecting an MP3 mode in the pocket-sized mobile communication device using the interface". For example, the Nokia 3300 Extended User's Guide at p.28 describes pressing the music key and selecting the music player from the menu. Furthermore, the Extended Guide teaches the user "[y]ou can listen to MP3 and AAC music files on the memory card in your phone with the Music player, or you can listen to the Radio. The special Music key on the top left of your phone enables you to quickly turn on and off both the Music player and the Radio." Nokia 3300 Extended User Guide at p.38.

265. The screenshot of the Nokia 3300 provided in Exhibit 6 for claim 1 element [c], taken by me, illustrates how one selects the music player application from the application menu.

266. I agree with the PTO examiner that Kokobu discloses this claim element. As discussed above for element [b], the portable phone disclosed in Kokobu is operable to process audio data, such as that in an MP3 music file. "In the portable telephone 100 shown in FIG. 10, an audio screen 122e is displayed on the main display unit 122, reproduction processing of audio data is carried out, and a plurality of the icons 10 are displayed on the sub-display unit 142." Kokubo at Col. 12:44-48. "For example, if, as shown in FIG. 10, the portable telephone 100 is used for listening to music, the audio player software is active, and the audio screen 122e is displayed on the main display unit 122, and a manually or automatically generated '♪' icon 10f is displayed on the sub-display unit 142." Kokubo at Col. 13:4-10.

267. **Element [d]** of claim 1 requires "selecting and playing a music file in the pocket-sized mobile communication device in the MP3 mode." "Listening to Music Press the Music key and select Music player." See Nokia 3300 Extended User's Guide at p.38 for complete instructions, as provided in the attached chart in Exhibit 6. I have taken two photographs which show the selecting and playing a music file on the Nokia 3300. . The first photograph illustrates how one selects the music player application from the application menu. The second photograph illustrates the music player application 6 seconds into playing a song by the name A Volte Il Cuore.mp3. See Exhibit 6.

268. Miyasaka also teaches selecting and playing a music file in the pocket-sized mobile communication device in the MP3 mode, for example see para. [119] and [135]. Again, as mentioned previously, Samsung admitted during prosecution that Miyasaka meets the limitation of claim 1.

269. **Element [e]** of claim 1 requires "switching the MP3 mode to a standby mode while the playing of the music file continues." The Nokia 3300 Extended User Guide discloses a standby mode including background music play. See, e.g., User Guide at p.22 for a detailed explanation of the standby mode, including an illustration of the standby screen showing the title of the music file track playing when the music player is on. The User Guide at p.99 notes that

"when the phone is in standby mode and the music player is on, you can control the player by simultaneously pressing the Music key and the appropriate function on the 4-way scroll key."

270. **Element [f]** of claim 1 requires "displaying an indication that the music file is being played in the standby mode." As described above, see Nokia 3300 device and the Extended User Guide, both of which teach displaying the title of the music track playing when the phone is in standby mode. See, e.g., Nokia 3300 Extended User Guide at p. 22 and p.99.

271. Miyasaka also teaches an indication that the music file is being played in the standby mode. "When only the music reproduction task is executed… the title of reproduced music, the singer, and others are displayed on the screen of the cell phone, as shown in FIG. 17A." Miyasaka at para. [135]. See also, Miyasaka at FIG. 17A, provided in the accompanying chart for element 1[f] at Exhibit 6. I note that Samsung has admitted that Miyasaka teaches at least the display of an indication that a music file is being played, as discussed above (see para. 6).

272. **Element [g]** of claim 1 requires selecting and performing at least one function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues. The Nokia 3300 and Extended User's Guide teach "selecting and performing at least one function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues." Nokia 3300 performs at least each of the following functions from the standby mode while the playing of the music file continues: messaging, stopwatch, and calendar functions. I have personally performed these tasks as selected from menu and confirm that the music file play continues. I have provided screenshots, taken by me, to show the message, stopwatch, and calendar functions on the screen. The music file was playing at each time the screenshots were taken. See claim 1[g], Exhibit 5.

273. I note that Samsung admitted during prosecution that this element was already in the prior art, specifically as disclosed in Miyasaka. See prosecution history summary above. See also, Miyasaka, at para. [023], [123], [136].

274. **Element [h]** of claim 1 requires and continuing to display the indication that the music file is being played while performing the selected function. Miyasaka teaches this element as reproduced in the accompanying chart in Exhibit 6, FIG. 17B, which indicates that the display shows an indication reading "Music reproduction B" even while the TV broadcast of a baseball game is being concurrently displayed. See also, Miyasaka at para. [136].

275. Dependent **claim 2** requires all the above limitations of claim 1, and further requires "wherein the displaying of the indication comprises displaying an icon." As the PTO Examiner noted in rejecting a previous iteration of claim 2, "[a]s to claim 2, KOKUBO teaches displaying of the indication comprises displaying an icon (col. 13, lines 4-10; col. 14, lines 17-31)." Office Action of November 9, 2009 at p.6.

276. Dependent **claim 7** requires all of the above limitations of claim 1, and further requires "wherein the function selected from the standby mode comprises a message function." The Nokia 3300 device and User Guide disclose this element as provided above in para. 272 and as shown for claim 1, element [g].

277. Dependent **claim 8** of the '711 patent requires all the elements of claim 1 and further requires "wherein the function selected from the standby mode comprises a phone-book function." The Nokia 3300 and User Guide disclose selecting a phone book function called "Contacts" and described at the Extended User Guide at pp.48-52. I have provided screenshots of the contacts function, taken by me, in the chart attached as Exhibit 6.

278. Independent **claim 9** is nearly identical to claim 1 and the claim limitations may be found in the identical locations as provided above for claim 1.

279. In addition to the elements required by claim 1, claim 9 also requires a "controller" for generating the music background play object of element [a]. The '711 specification states that "[a] controller 110 controls the overall operation of the portable terminal." '711 patent at Col. 3:52-53. I have inspected the Nokia 3300 phone and verified that it comprises a processor, which the person of ordinary skill would recognize controls the overall

operation of the portable terminal. Further, Kokubo teaches a control unit 115 that functions as a controller, as described in Kokubo at Col. 6:33-64, and Miyasaka teaches "[a] multitask control device for causing a processor (10) to execute concurrently a plurality of tasks" that meets this added limitation of claim 9. See, e.g., Miyasaka at Abstract and para. [0015]-[0018] and FIGS. 1, 5, 9, and 13-15.

280. Claim 9 also requires "a display unit for displaying an indication that the music file is being played in the standby mode." The Nokia 3300 and User Guide disclose a display unit as described for the interface limitation of claim 1, element [b]. The cited disclosure for that element is incorporated here, and provided in the chart attached as Exhibit 6.

281. Dependent **claim 10** requires the same elements as claim 9, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claim 2 and is disclosed by Kokubo as discussed above for claim 2 See paragraph 275.

282. Dependent **claim 15** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a message function." This additional limitation is the same as that required by claim 7 and is disclosed by the Nokia 3300 and User Guide as discussed above for claim 7. See paragraph 276.

283. Dependent **claim 16** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a phonebook function." This additional limitation is the same as that required by claim 8 and is disclosed by the Nokia 3300 and User Guide as discussed above for claim 8. See paragraph 277.

284. Independent **claim 17** requires similar limitations as claim 9, but does not require the "selecting and performing at least on function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues" limitation, nor does it require the "continuing to display the indication that the music file is being played while

performing the selected function" limitation of claim 9. All elements of claim 17 are otherwise disclosed as described above for claim 9 and claim 1.

285. Dependent **claim 18** requires the limitations of claim 17 above, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claims 2 and 10 and is disclosed by the Kokubo patent as discussed above for those claims.

8. Motivation to combine the above references as claimed

286. The person of ordinary skill in the art would have been motivated to combine the Nokia 3300 device and accompanying manual with the teachings of Miyasaka because while the Nokia 3300 does display an indication that the music device is being played, it does not necessarily continue to display the indication while performing multitasking. As Samsung admitted during prosecution (see para. 123), Miyasaka teaches continuing to display the indication that the music file is being played while performing the selected function. To the extent this element is deemed to be missing from the disclosure of the Nokia 3300 device and accompanying manual, the advantage of indicating to the user that a music file is being played while performing another selected function would have been evident to the person of ordinary skill by 2005. Displaying a music play indication during multitasking helps to remind the user that a music file is playing in the background, for example if the user has turned the volume down and forgotten that the file is playing. The user may wish to turn off the music play application in order to preserve battery life in the mobile device.

287. The person of ordinary skill in the art would have been motivated to combine the Nokia 3300 device and accompanying manual with the teachings of Kokubo because Kokubo teaches the use of an display icon indicating music play (see above at para. 107-109), which would be useful for the user of a multitasking device such as the Nokia 3300 mobile phone for the same reasons as described above for the combination of the Nokia 3300 art and Miyasaka.

288. Wong, Shaffer, or Mahmoud would also have been obvious to combine with the disclosure of the Nokia 3300 and one of Kokubo or Miyasaka because these references motivate

the ordinary artisan to program the music player application with at least one applet. For a discussion of the general advantages of applets and the specific advantages explicitly recited by these references, please see paragraphs 175-189 above. Like the Sony W800i and K700i phones, the Nokia 3300 also supports Java J2ME, making it all the more likely that an applet would have been straightforward to implement in this phone. See Nokia 3300 Extended User Guide at p.18.

9. The '711 claims are obvious over Miyasaka or Kokubu, in view of Forsyth and any of the Wong patent, Shaffer patent, or Mahmoud article

289. It is my opinion that Claims 1, 2, 7-10, and 15-18 of the '711 patent would have been obvious to a person of ordinary skill in the art in 2005 because (1) all elements of claim 1 were present in the prior art, specifically in a combination of Kokubo, Forsyth, and any of Wong, Shaffer, or Mahmoud, or alternatively in a combination of Miyasaka, Forsyth, and any of Wong, Shaffer, or Mahmoud. To the extent that Samsung suggests that, contrary to the findings of the Examiner, the Kokubo reference does not teach "switching the MP3 mode to a standby mode while the playing of the music file continues" as required by claims 1, 9, and 17, it is my opinion that this element is further taught by the Forsyth patent as detailed below.

290. Independent claim 1 would have been obvious to the person of ordinary skill in the art because all claim elements but the requirement that the music background play object application module include at least one applet were present in the Kokubo patent or the Miyasaka publication. However, it would have been obvious to the person of ordinary skill in the art in 2005 that including an applet in an application module controlling music background play function would be desirable for numerous reasons as discussed below in para. 175-189.

291. As described above at para. 123, Samsung admitted in the Accelerated Examination Support Document filed July 16, 2007 that the Miyasaka reference teaches at least the following elements:

- A multi-tasking method in a pocket-sized mobile communication device, the method comprising:

- Selecting and playing a music file in the pocket-sized mobile communication device
- Displaying an indication that the music device is being played
- Selecting and performing at least one function of the pocket-sized mobile communication device while the playing of the music continues
- Continuing to display the indication that the music file is being played while performing the selected function
- Selecting a message function as required by claims 7 and 15
- A controller for selecting and playing a music file in the pocket-sized mobile communication device and selecting and performing at least one function of the pocket-sized mobile communication dev ice while the playing of the music file continues as required by asserted independent claim 9
- A multi-tasking apparatus in a pocket-sized mobile communication device and a display unit for displaying an indication that the music file is being played as relates to independent claim 17

292. Claim 1 requires "a multi-tasking method in a pocket-sized mobile communication device including an MP3 playing capability." Miyasaka meets this limitation, as discussed above at para. 257 and by Samsung's own admission during prosecution. Kokubo also meets this limitation, as discussed above at para. 193.

293. **Element [a]** of claim 1 recites "generating a music background play object, wherein the music background play object includes an application module including at least one applet." Miyasaka and Kokobu both teach this limitation. See para. 261 above and the chart in Exhibit 7 for the specific disclosure meeting this limitation in Miyasaka. See para 196 above for the specific disclosure in Kokubu.

294. Forsyth discloses a standby or "idle" screen for use in mobile telephones and similar devices, which are capable of MP3 music file play. For example, Forsyth teaches that "[i]f the mobile telephone device is also a music download platform (e.g. downloads from mp3 music sites or is a digital radio receiver), then information in the idle screen can relate to the currently played music track (e.g. artist information, track information, concert dates, links to e-

commerce functionality such as buying CDs)." Forsyth application at para. [123]. See also, Forsyth at para. [002] as provided in the chart at **Exhibit 7**.

295. See para. 175-189 above specifying relevant disclosure in the references disclosing the use of applets for music play functions, including Wong, Shaffer, and Mahmoud.

296. **Element [b]** of claim 1 requires "providing an interface for music play by the music background play object." Miyasaka teaches an interface for music play by a music background play object at para. [0112] and [0113], referencing FIG. 11, provided in the accompanying chart at Exhibit 7.

297. Kokubo, as described above at para. 197, discloses this limitation. See also, Kokubo at 2:34-41, 12:44-48, 13:4-10, and FIGS. 1A, 1B, 3, 5, 7, and 10-13.

298. **Element [c]** of claim 1 requires selecting an MP3 mode in the pocket-sized mobile communication device using the interface. Miyasaka and Kokubo both teach this element. Miyasaka, for example, teaches a "music reproduction task" or mode on the disclosed mobile phone. See, e.g., Miyasaka at para. [119] and [135].

299. Kokubo discloses this element as shown above at para. 198. See, e.g., Kokubo at 12:44-48 and 13:4-10.

300. **Element [d]** of claim 1 requires selecting and playing a music file in the pocket-sized mobile communication device in the MP3 mode. At the same locations as the above cites, both Miyasaka and Kokubo teach selecting and playing a music file on a mobile phone. See accompanying chart at Exhibit 7.

301. **Element [e]** of claim 1 requires switching the MP3 mode to a standby mode while the playing of the music file continues. Forsyth teaches this element at para. [002] and [123]. The mobile phone described has an idle or standby screen that is the default setting when the user is not using applications such as the messaging, but music play continues in the idle state.

302. **Element [f]** of claim 1 requires displaying an indication that the music file is being played in the standby mode. Forsyth discloses this limitation at para. [123] because it

explicitly describes displaying an indication on the idle (standby) screen ("information in the idle screen can relate to the curretly played music track.")

303. Miyasaka discloses this element as described above at para. 271.

304. Kokubo discloses this element as described above at para. 204.

305. **Element [g]** of claim 1 requires selecting and performing at least one function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues.

306. Kokubo, as discussed at para. 205 above, discloses this limitation.

307. Miyasaka, as discussed at para. 273 above, discloses this limitation.

308. **Element [h]** of claim 1 requires and continuing to display the indication that the music file is being played while performing the selected function. Again, for Kokubo, see disclosure cited in para. 207 and Exhibit 7.

309. Miyasaka, as discussed at para. 274 above, discloses this limitation.

310. Dependent **claim 2** requires all the above limitations of claim 1, and further requires "wherein the displaying of the indication comprises displaying an icon." As the PTO Examiner noted in rejecting a previous iteration of claim 2, "[a]s to claim 2, KOKUBO teaches displaying of the indication comprises displaying an icon (col. 13, lines 4-10; col. 14, lines 17-31)." Office Action of November 9, 2009 at p.6. :

311. Dependent **claim 7** requires all of the elements of claim 1 and further requires "wherein the function selected from the standby mode comprises a message function." As discussed above, Samsung has already admitted this limitation is present in at least the Miyasaka reference, and the limitation is further taught by Kokubo as noted by the Examiner in the Office Action of November 9, 2009 (discussed above at para. 209).

312. **Claim 8** of the '711 patent requires all the elements of claim 1 and further requires "wherein the function selected from the standby mode comprises a phone-book function." As noted by the Examiner in the Office Action of November 9, 2009 (discussed

above at para. 209, Kokubo discloses selecting a phone book function from the standby mode. Forsyth further discloses this function as cited below:

313. Forsyth discloses a "contacts" application on the mobile phone device at para. [002], which would be understood by the person of ordinary skill in the art as a "phone book."

314. Independent **claim 9** is nearly identical to claim 1 and the claim limitations may be found in the identical locations as provided above for claim 1.

315. In addition to the elements required by claim 1, claim 9 also requires a "controller" for generating the music background play object of element [a]. The '711 specification states that "[a] controller 110 controls the overall operation of the portable terminal." '711 patent at Col. 3:52-53. Kokubo teaches a control unit 115 that functions as a controller, as described in Kokubo at Col. 6:33-64, and Miyasaka teaches "[a] multitask control device for causing a processor (10) to execute concurrently a plurality of tasks" that meets this added limitation of claim 9. See, e.g., Miyasaka at Abstract and para. [0015]-[0018] and FIGS. 1, 5, 9, and 13-15.

316. Claim 9 also requires "a display unit for displaying an indication that the music file is being played in the standby mode." Miyasaka, Kokobu, and Forsyth each discloses a display unit, as shown in the accompanying chart in Exhibit 7.

317. Dependent **claim 10** requires the same elements as claim 9, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claim 2 and is disclosed by Kokubo as discussed above for claim 1.

318. Dependent **claim 15** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a message function." This additional limitation is the same as that required by claim 7 and is disclosed by each of Miyasaka, Kokubo, and Forsyth as discussed above for claim 7.

319. Dependent **claim 16** requires the same elements as claim 9, but additionally requires "wherein the function selected from the standby mode comprises a phonebook

function." This additional limitation is the same as that required by claim 8 and is disclosed by each of Miyasaka, Kokubo, and Forsyth as discussed above for claim 8.

320. Independent **claim 17** requires similar limitations as claim 9, but does not require the "selecting and performing at least on function of the pocket-sized mobile communication device from the standby mode while the playing of the music file continues" limitation, nor does it require the "continuing to display the indication that the music file is being played while performing the selected function" limitation of claim 9. All elements of claim 17 are otherwise disclosed as described above for claim 9 and claim 1.

321. Dependent **claim 18** requires the limitations of claim 17 above, and further requires "wherein the indication comprises an icon." This additional limitation is the same as that required by claims 2 and 10 and is disclosed by the Kokubo patent as discussed above for those claims.

10. Motivation to combine the above references as claimed

322. As discussed above, the Kokubo reference was determined by the examiner during prosecution to teach the claimed limitations of the '711 patent but did not teach a standby screen or generating a music background play object that includes an application including at least one applet. Combining the Kokubo reference with the Forsyth reference would have been motivated by the recognizable advantages of having a standby or "idle" screen on a multitasking mobile device. Forsyth includes multiple potential applications which can be executed from the standby screen on a mobile device, including MP3 music file functionality. See, e.g., Forsyth at ¶¶ 002 and 123. By 2005, the person of ordinary skill would appreciate the advantage that a standby screen might have, including, as explicitly recited by Forsyth, the ability to present information of interest to the user on a screen that requires no navigation to select it. See Forsyth at Abstract.

323. The person of ordinary skill in the art would also have found it obvious to combine Miyasaka with Forsyth for much the same reasons as above; Forsyth motivated the used

of a standby screen on a mobile phone device and would have provided the device disclosed in Miyasaka the opportunity to provide information to the user without navigating to any further screens, as described in the paragraph above.

324. Wong, Shaffer, or Mahmoud would also have been obvious to combine with the disclosure of either Kokubo or Miyasaka because these references motivate the ordinary artisan to program the music player application with at least one applet. For a discussion of the general advantages of applets and the specific advantages explicitly recited by these references, please see paragraphs 175-189 and the discussions of these references in Section IX A above.

X. Secondary Considerations

325. I understand that so-called "secondary considerations" or "objective indicia of non-obviousness" may provide evidence that a claimed invention was not obvious at the relevant date. These secondary considerations of non-obviousness include the following:

- Commercial success
- Skepticism of experts at the time
- Evidence of copying
- Evidence that others tried and failed to solve the same problem
- Unexpected results
- Evidence of acclaim by others or other recognition
- Evidence of a long-felt need in the art

326. I have been informed by counsel that Samsung provided supplemental interrogatory responses on "secondary considerations" topics just two days ago, [redacted]

[redacted]

327. [redacted]

328. I am not aware of any evidence showing that commercial success of products practicing the claims of the '711 patent, if any, is attributable to the use of a "music background play object including an application module including an applet. Further, I am not aware of evidence showing that the commercial success of products incorporating the '711 technology, if any, was because of the ability of these devices to play music files in the background while multitasking. [redacted]

329. It is my opinion that no expert during the relevant time period would have doubted that it would have been relatively straightforward to program a mobile device with background music play functionality allowing the user to multitask while playing an MP3 file. In fact, as I discussed above, mobile phones were capable of background music play while multitasking since at least 2003. With or without programming that includes an applet, I see no reason why experts would have doubted the feasibility of such technology. I have detailed elsewhere in this report, multitasking performed by software instead of by additional processors was well known by 2005 and would not have been considered difficult by engineers at that time.

[redacted]

XII. Trial Exhibits

351. If called as a witness at trial, I may rely on visual aids and demonstrative exhibits that demonstrate the bases of my opinions. Examples of these visual aids and demonstrative exhibits may include, for example, claim charts, patent drawings, excerpts from patent specifications, file histories, interrogatory responses, deposition testimony and deposition exhibits, as well as charts, diagrams, videos and animated or computer-generated video.

352. Other than as referred to in this report, I have not yet prepared any exhibits for use at trial as a summary or support for the opinions expressed in this report, but I expect to do so in accordance with the Court's scheduling orders.

XIII. Previous Testimony

353. On November 14, 2011, I submitted a declaration in support of Apple's proposed claim construction of the term "applet." I testified on the subject matter of that declaration in a deposition on December 6, 2011.

354. Apart from my previous testimony above, I have not previously testified nor served as an expert in a litigation.

XIV. Supplementation of opinions

355. I reserve the right to supplement my analysis in light of any critique of my report or alternative opinions advanced by or on behalf of Samsung.

Dated: 3/22/2012

Tony D. Givargis, Ph.D.