EXHIBIT E

UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA
SAN JOSE DIVISION

APPLE INC., a California corporation *Plaintiffs*,		
v.	§	
	§	
SAMSUNG ELECTRONICS CO., LTD., a	§	
Korean business entity, SAMSUNG	§	CIVIL ACTION NO. 11-CV-01846-LHK
ELECTRONICS AMERICA, INC., a New	§	
York corporation, and SAMSUNG	§	
TELECOMMUNICATIONS AMERICA,	§	
LLC, a Delaware limited liability company	§	
	§	
Defendants.	§	

DECLARATION OF TONY GIVARGIS, PH.D.
IN SUPPORT OF APPLE'S PROPOSED CLAIM CONSTRUCTION
FOR U.S. PATENT NO. 7,698,711

I. Introduction

1. I have been retained as an expert in this case by Apple Inc. ("Apple"). In this Declaration I provide my opinions regarding the interpretation of the term "applet" as used in the claims of U.S. Patent No. 7,698,711 to Jeong ("the '711 patent").

II. Qualifications

2. 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.

3. Since 2001, I have been a member of the Department of Computer Science faculty at the University of California, Irvine ("UC-Irvine"). 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.

4. My research is 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. An inherent 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 intended 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 graduated four Ph.D. students and am currently directing 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.

5. As a professor, I regularly teach both at the graduate and undergraduate levels. At UC-Irvine, among computer science and engineering students, I am 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.

6. 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, archived, and ACM/IEEE-sponsored venues. 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 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.*

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

8. I am attaching as Exhibit A my updated curriculum vitae setting forth my qualifications and publications.

III. Materials Reviewed

9. My opinions expressed in this declaration are based on my review of the following materials:

   A. The ‘711 patent

   B. The prosecution history of the ‘711 patent

   C. The prosecution history of the parent application of the ‘711 patent (U.S. Application No. 11/390,338)

D. Apple's LPR 4-2 disclosure of proposed claim construction for "applet" and materials cited in support

E. Samsung's LPR 4-2 disclosure of proposed claim construction for "applet" and materials cited in support

10. I also based this opinion on my professional and academic experience including in the field of embedded systems and programming for multitasking operations.

11. If asked, I may testify on these materials and experience.

IV. Understanding of the Law

12. I have been informed and understand that claim construction is the process of determining a patent claim's meaning. I also have been informed and understand that the proper construction of a claim term is the meaning that a person of ordinary skill in the art (*i.e.*, the technical field to which the patent relates) would have given to that term at the time when the patent's application was filed.

13. I understand that, for claim construction, one must focus on the claim terms in the context of the claim as a whole, interpreting the claim language as it ordinarily would be understood. After the claim language, the most important sources to consider are the patent specification, followed by the prosecution history. I understand that, collectively, these sources—the claim language, specification, and prosecution history—are called "intrinsic evidence."

14. I further understand that a patentee may act as his own lexicographer and provide definitions in the specification. If a patent contains a clear definition of a term, then the term should be construed in accordance with that definition.

15. I further understand that actions taken by the patent owner can affect the constructions of the claim terms. For example, if the patent owner distinguishes the claims from prior art during prosecution, it would generally be incorrect to construe the claims so as to cover the distinguished material.

16. In addition, claim construction may take into account dictionaries, technical references, and other information—called "extrinsic evidence"—that would have been available to those skilled in the art at the time when the patent's application was filed. I understand that the law considers extrinsic evidence to be less reliable than intrinsic evidence, and that extrinsic evidence cannot change the ordinary meaning of the claim language.

V. Person of Ordinary Skill in the Art

17. The '711 patent was filed in the U.S. Patent and Trademark Office on July 16, 2007 and claims priority to Korean patent application 10-2005-0079921, filed on August 30, 2005. Accordingly, I understand I am to interpret the claim terms in

the ‘711 patent from the perspective of the person of ordinary skill in the art as of August 30, 2005.

18. For the '711 patent, a person of ordinary skill in the art in 2005 would have at least a bachelor's degree in computer science/engineering or similar discipline and several years' 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 Background

19. In some instances in computer programming, it is desirable for a first program or application to run within the context of a second application (or “host”), where the host application interprets and executes the instructions of the first program. This can be advantageous, for example, so that the host application can insulate or protect a user’s computer from a potentially suspicious first application that is obtained externally, *e.g.*, on the internet. Another advantage is that the first application or program can be accessible to a diverse user base and be implemented in essentially the same form across multiple different platforms.

20. That is, in software systems where a first application executes within the context of a second “host” application, the first application 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. For example, in 2005, a video game program could run on a host web browser, such as Microsoft Internet Explorer running on a Windows-based PC. The same video game program could also run on a different host browser, such as Netscape on a laptop running the Linux operating system.

21. Java provides a common example of such programs (today as well as in 2005). Java is a computer programming language 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.

22. Leading up to 2005, mobile phone manufacturers increasingly produced Java-enabled devices. Java applications running on mobile devices had access to the multimedia capabilities of the device, including audio and video playback. For example, major phone manufacturers such as Sony Ericsson, Nokia, and others marketed Java-enabled mobile phones by 2005. Third-party developers commonly developed programs for the Java-enabled mobile phone market.

VII. The ‘711 patent disclosure

23. The ‘711 patent describes a “portable terminal,” such as a cellular telephone, which can be used by the operator to listen to an MP3 music file while simultaneously working in another application, such as a messaging function. *See*, *e.g.*, ‘711 patent at Abstract.

24. In the “Background of the Invention” section of the ‘711 patent, the specification describes the purported problem in the art being addressed by the patent:

> Generally, a portable terminal having an MP3 music play function controls and plays an MP3 music file using a separate player. For example, a user may select an MP3 music play function from menus of a portable terminal for listening to the music. The user may also selectively use control-related functions such as play, pause, repeat, and terminate the MP3 music play function.
>
> However, the user cannot simultaneously work on several menus of the portable terminal while listening to the music using the conventional MP3 music play function. In other words, the user cannot use the other functions of scheduling, picture viewing, or game menu among others while listening to the music.
>
> To address this problem, a control processor is added to manage only MP3 music play, resulting in an increase in cost and an increase in the complexity of hardware and software configurations.
>
> Accordingly, there is 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.

‘711 patent at Col. 1:32-51

25. The “Summary of the Invention” section goes on to state:

> An aspect of exemplary embodiments of the present invention is to address at least the above problems and/or disadvantages and to provide at least the advantages described below. Accordingly, an aspect of exemplary embodiments of the present invention is to provide a multi-tasking apparatus and method in a portable terminal, in which menu functions of the portable terminal can be implemented while continuing to play a music file.

‘711 patent at Col. 1:55-62.

26. The ‘711 patent goes on to discuss multi-tasking using “a music background play object.”

> According to another aspect of an exemplary embodiment of the present invention, a multi-tasking method in a portable terminal is provided. Where a music background play object is generated in a standby mode. The music background play object acquires a task for music background play and provides a control interface for music play. The music file is played through multi-tasking in the standby mode using the music background play object upon selection of a music play mode in the portable terminal. A menu function of the portable terminal in the music play mode is selected. The portable terminal switches to a mode of the selected menu of the portable terminal and implements a corresponding function while continuing to play the music upon selection of the menu function of the portable terminal. The portable terminal switches to the music play mode upon termination of the mode of the selected menu of the portable terminal.

'711 patent at Col. 2:26-41.

27. The '711 further describes "a music background play object" with respect to the flowchart illustrated in FIG. 2:

> Referring to FIG. 2, the controller 110 of the portable terminal generates an MP3 music background play object in the standby mode in step 201 to play an MP3 music file as background music. The MP3 music background play object generated in step 201 internally acquires a task for MP3 music background play, returns to the standby mode, and provides a control interface to allow other applications to transmit commands for music play and control through the MP3 music background play object.
>
> When an MP3 music play command is selected in the portable terminal, the controller 110 senses the selection and executes an MP3 music play application in step 202. The controller 110 transmits the MP3 music play command to the activated MP3 music background play object in the standby mode through an interface for an execution command provided by the MP3 music background play object. The MP3 music background play object switches to an MP3 music play mode by multi-tasking in step 203.

'711 patent at Col. 4:52-Col. 5:2.

28. The only recitation of the term "applet" in the specification is as follows:

> FIG. 1 is a block diagram of a portable terminal according to an exemplary embodiment of the present invention, in which an MP3 music control processor is not included. Application modules of the portable terminal include at least one **applet** and each of the

application modules, that is each menu of the portable terminal, independently performs multi-tasking.

'711 patent at Col. 3:8-14 (bold emphasis added).

VIII. The '711 patent claims and file history

29. The '711 patent issued with 20 claims, including 3 independent claims (claims 1, 9, and 17). I understand that claims 1, 2, 7-10, and 15-18 are being asserted by Samsung in this litigation. Representative claim 1 is reproduced below (letter designations (a-h) added and the term "applet" highlighted in bold for clarity):

> 1. A multi-tasking method in a pocket-sized mobile communication device including an MP3 playing capability, the multi-tasking method comprising:
>
> (a) generating a music background play object, wherein the music background play object includes an application module including at least one **applet**;
>
> (b) providing an interface for music play by the music background play object;
>
> (c) selecting an MP3 mode in the pocket-sized mobile communication device using the interface;
>
> (d) selecting and playing a music file in the pocket-sized mobile communication device in the MP3 mode;
>
> (e) switching the MP3 mode to a standby mode while the playing of the music file continues;
>
> (f) displaying an indication that the music file is being played in the standby mode;
>
> (g) 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;
>
> (h) and continuing to display the indication that the music file is being played while performing the selected function.

30. In addition to claim 1, the term "applet" is recited in claims 9 and 17 claiming "a controller for generating a music background play object, wherein the music background play object includes an application module including at least one applet."

IX. Disputed claim term of the '711 patent

31. The following term is disputed between the parties:

A. "applet"

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."

1. Meaning to One of Ordinary Skill in the Art

32. A person of ordinary skill in the art in 2005 would have understood the term "applet" in the claims of the '711 patent to mean "an operating system-independent computer program that runs within an application module."

2. Intrinsic Evidence

33. The language of claims 1, 9, and 17 requires "an application module including at least one applet." One of ordinary skill in the art would therefore understand that the claimed applet runs within an application module.

34. Neither the '711 specification nor the file history provides a definition of the term "applet." The only recitation of the term "applet" in the specification is as follows:

> FIG. 1 is a block diagram of a portable terminal according to an exemplary embodiment of the present invention, in which an MP3 music control processor is not included. Application modules of the portable terminal include at least one **applet** and each of the application modules, that is each menu of the portable terminal, independently performs multi-tasking.

'711 patent at Col. 3:8-14 (bold emphasis added).

35. The prosecution file history of the '711 patent contains several mentions of the term "applet" but does not provide a definition. Remarks made by the examiner and applicant including the term "applet" are provided in the following paragraphs.

36. "Examiner suggested to further include the definition 'a music background play object' as 'wherein the music background play objects including an application

module includes at least one **applet**' as argued during the interview to distinct [sic] from the icon as taught by KOKUBO." U.S. Patent Application No. 11/778,466, Examiner's Interview Summary of December 16, 2009, see Continuation Sheet (bold emphasis added).

37. "The Office did suggest, however, that the inclusion of a limitation further defining the music background play object would distinguish over the prior art of record, though not necessarily be allowable depending on the results of a further search. Specifically, the Office suggested including a limitation indicating that the music background play object includes an application module including at least one **applet**. Applicant appreciates the Office's suggestion and has amended the independent claims as suggested." U.S. Patent Application No. 11/778,466, Applicant's December 8, 2009 Arguments/Remarks Made in an Amendment at pp. 6-7 (bold emphasis added).

38. "For the specific invention as claimed, a music background play object, wherein the music background play object includes an application module including at least one **applet**, is included such that an MP3 file can be played in the background while other menu tasks can be executed by the user." *Id*. at p.7 (bold emphasis added).

39. "By use of the music background play object, which is an application module including at least one **applet** as discussed with reference to para. [0018], the terminal is able to perform multi-tasking. That is, by generating the application module of the music background play object, the music background play object provides an interface for the playing of music, specifically the selecting of an MP3 mode. At the same time, the user is able to execute other menu functions of the device and thus multi-task using the device. It is Applicant's contention that independent claims 1,9 and 17 are allowable based on the unique use of the music background play object, wherein the music background play object includes an application module including at least one **applet**, alone, and not based on the use of the music background play object in a standby or any other mode. That is, none of the prior art discloses a music background play object, wherein the music background play object includes an application module including at least one **applet** in any mode of a device. Accordingly, Applicant believes that the claims are in condition for immediate allowance." *Id*. at pp. 7-8 (bold emphasis added).

40. "As suggested by the Office during the interview, this clarifying limitation is not disclosed, taught or suggested by Kokubo. Rather, as acknowledged by the Office in the outstanding rejection, Kokubo merely discloses the generating of 'an icon corresponding to a task (application software)' *col. 2, lines 34-39*; see also col. 13, lines 8-10 ('manually or automatically generated music [music note symbol] icon 10f is displayed.' The generating of the icon by Kokubo is not a disclosure of generating a music background play object, wherein the music background play object includes an application module including at least one **applet**. That is, Kokubo makes no disclosure that the icon includes an application module, or that the application module includes at least one **applet** as instantly

claimed." *Id*. at pp. 9-10 (italics in original) (bold emphasis added). *See also*, '711 patent claims 1, 9, and 17.

41. These passages from the file history reinforce the clear language of the claims and specification stating that the claimed applet runs within an application module.

3. Extrinsic Evidence

42. The *McGraw-Hill Dictionary of Scientific and Technical Terms* (6th Ed., 2003) at p.124 defines "applet" as "a small program, typically written in Java." This definition supports Apple's proposed construction because persons of ordinary skill in the art in 2005 would commonly associate the term "applet" with an application written in the Java programming language, and Java was well-known to be operating system-independent, as described above.

43. The *Dictionary for Library and Information Science* (1st Ed., 2004) at p.34 defines "applet" as "a small application program written in the Java programming language developed by Sun Microsystems for distribution over the Internet. Applets run on any Java-enabled Web browser independent of platform (Windows, Macintosh, UNIX, etc.)." This definition again supports Apple's proposed construction because persons of ordinary skill in the art in 2005 understood "applets" as commonly being written in the Java programming language, which is characterized by programming independent of the platform, including the operating system.

44. The *Java Developer's Resource* (1997) by Eliotte Harold ("the Harold reference") at p.11 states that "[w]hat's most special about Java in relation to other programming languages is that it lets you write special programs called applets that can be downloaded from the Internet and played safely within a Web browser." The discussion in this text supports Apple's construction because it reflects the common understanding of persons of ordinary skill in the art in 2005 that applets, typically written in the operating system-independent Java programming language, are executed within an application module, e.g., a Web browser. The Web browser or other application module provides the execution environment for the applet.

45. The Harold reference at p.12 further explains how applets can be independent of the host platform: "Java solves the problem of platform independence by using byte code…. Java programs that have been compiled into byte code still need an interpreter to execute them on any given platform. The interpreter reads the byte code and translates it into the platform's native language on the fly. The most common such interpreter is Sun's program java (with a little j). Since the byte code is completely platform independent, only the interpreter and a few native libraries need to be ported to get Java to run on a new computer or operating system."

46. Thus, the above excerpt in the Harold reference further supports Apple's proposed construction of "applet" as being "operating system-independent." Persons of ordinary skill in the art in 2005 would have understood Java applications, including applets, as being processor and operating system "agnostic" or independent. Specifically, Java applications, including applets, execute within a standardized execution environment. An "interpreter," designed according to the standardized execution environment and usually a component of the host application module, translates the instructions of the Java applications, including applets, to those of the host platform and operating system. Thus, applets are "independent" of the operating system because they can rely upon the interpreter in the host application module to translate their instructions for them; they do not interact directly with the operating system and instead, in the context of the '711 patent, run within the application module.

47. Further, the Harold reference at p.12 describes the security advantage of applets as programs running within an application module: "Java solves this [security] problem by severely restricting what an applet can do. A Java applet cannot write to your hard disk without your permission. It cannot write to arbitrary addresses in memory and thereby introduce a virus into your computer. It cannot crash your system." *See also*, e.g., Harold at pp. 9-34. Thus, persons of ordinary skill in the art in 2005 would understand "applet" to be a program that runs in the context of another application module, such as a Web browser. The Web browser provides the execution environment for the applet and restricts its access to a user's computer resource and private data.

48. *Java: An Introduction to Computer Science & Programming* (3rd Ed., 2004) by Walter Savitch ("the Savitch reference") explains that "[t]he word *applet* sounds as though it might refer to a small apple, but it is supposed to sound like a small application. Thus, applets are just 'little Java programs,' in some sense of the word *little.* However, the character of applets comes not from their size, but from how and where they are run. Applets are Java programs that can be displayed on a Web site and viewed over the Internet. They can also be run on your local computer, without any connection to the Internet." Savitch at p.797 (italics in original). *See also, e.g.*, Savitch at Chapter 1, pp. 3-37, and Chapter 13, pp. 795-821. The Savitch reference supports Apple's proposed construction because it confirms that the person of ordinary skill would understand that applets have the characteristics of Java-based programs, including running within an application module independently of the operating system, as described in other references cited in this declaration.

49. In *IBM e-server pSeries* (12th Ed., 2004) by Hoskins, J. and Bluethman, R. *Exploring* ("the Hoskins reference"), "Java is an object-oriented programming environment that operates independent of any operating system or microprocessor. Java programs, called applications or applets, can be entirely developed using the compiler, debugger, and applet viewer tools provided in IBM's implementation for the AIX for Java development environment. (C and C++ compilers and tools are not needed to create/run Java-based applets.) The same applets can be

dynamically transmitted over a network and run on any client that has been enabled for Java. Because applet execution is platform independent, an applet developed with the AIX 5L tools can be executed on any Java-enabled platform (for example, Solaris)." Hoskins reference at p.226.

50. The Hoskins reference as cited above supports Apple's proposed construction of applet because it shows that persons of ordinary skill in the art in 2005 would define an "applet" as an application which "operates independent of any operating system." Further, Hoskins notes that "applet execution is platform independent," again supporting Apple's proposed construction that "applet" would have been understood to be a computer program that runs independent of the operating system. Java applications, including applets, are designed to execute within a standardized execution environment. A Java-enabled device provides an environment that translates the instructions of the Java applications, including applets, to those of the host platform and operating system.

51. Further academic publications prior to 2005 support Apple's proposed construction of "applet" as including operating system independence. *See*, e.g., "Healy, M.R., Berger, D.E., Romero, V.L., Aberson, C.L., & Saw, A. Evaluating JAVA applets for teaching on the Internet. *Proceedings of the Scuola Superiore G. Reis Romoli Advances in Infrastructure for e-Business, e-Education, e-Science, and e-Medicine on the Internet International Conference*. (2002) at p.1: "Java applets are computer applications designed for the Internet. Applets are platform-independent, meaning that they can run on any operating system that has a Java Virtual Machine to translate applet bytecodes into appropriate platform-dependent instructions." *See also, e.g.*, Healy at pp. 1-5, available online at: http://ccdl.libraries.claremont.edu/cdm4/item_viewer.php?CISOROOT=/irw&CISOPTR=432 .

52. Operating system-independence was an understood characteristic of applets in the art by 2005. As a further example, "*Interactive Programming with Java Applets*" (2005) by Elizabeth Boese ("the Boese reference") at p.8 notes that "Java is platform independent because the source code is compiled to bytecode, and it's the bytecode that can be used on any platform (operating system)."

53. The Boese reference further discusses at p.9: "There are two different types of Java programs that we can create: applets and applications. Applets are Java programs embedded into a web page. Applications are stand-alone programs that can be run by themselves." The Boese reference thus supports Apple's proposed construction of "applet" as a program "running within an application module" because applets, unlike applications, are not stand-alone programs and cannot run by themselves. *See also*, Boese at Chapter 1, pp. 7-20 (available online at http://books.google.com/books?id=mEC7H9WxXHEC&pg=PA8&dq=applets+are+operating+system+platform+independent&hl=en#v=onepage&q&f=false)

54. Apple's proposed construction of "applet" is further supported by *Web Technologies TCP/IP Architecture, and Java Programming*" (2nd Ed., 2002) by

Godbole, A. S. and Kahate, A. ("the Godbole reference") at p.524: "[b]y virtue of the Java heritage, applets are platform independent." (available online at http://books.google.com/books?id=uEufGycOJRsC&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=true). Thus, a person of ordinary skill in the art in 2005 would understand "applets" to be platform-independent, which necessarily includes "operating system-independent."

4. Conclusion

55. Based on the above intrinsic and extrinsic evidence, it is my opinion that a person of ordinary skill in the art in 2005 would have interpreted the term "applet" in the '711 patent claims according to Apple's proposed construction as "an operating system-independent computer program that runs within an application module."

X. Samsung's Proposed Construction of "Applet"

56. I have reviewed Samsung's Patent Local Rule 4-2 Disclosures, including Exhibit A at p.51. I understand that Samsung has proposed "applet" should be construed as "a small application designed to run within another program."

57. As support for its proposed construction, Samsung cites to the *Wiley Electrical and Electronics Engineering Dictionary* (2004), which offers the same definition. Samsung further cites to the single reference to "applet" in the '711 patent specification at Col. 3:10-14, as discussed above in this declaration.

58. It is my opinion that a person of ordinary skill in the art in 2005 would not have construed "applet" as broadly as proposed by Samsung. As an initial matter, the '711 patent specification and claims explicitly require a narrower reading in accordance with Apple's proposed construction. For example, as discussed above, each of claims 1, 9, and 17 requires "an application module including at least one applet." Based on the claim language alone it is clear that the claimed applet must run within an application module. Furthermore, the only reference to applet in the specification (at Col. 3:10-12) states that "[a]pplication modules of the portable terminal include at least one applet," confirming that the claimed applets run specifically within "application modules" and not simply any "program" in general as proposed by Samsung. As discussed above, the file history of the '711 patent also includes numerous statements reinforcing that the claimed applet runs within an application module.

59. It is further my opinion that "applets" would have been understood by a person of ordinary skill in the art in 2005 to be operating system-independent as required by Apple's proposed construction. As shown above in numerous supporting references, an "applet" was widely understood as a program that runs independently of the host operating system and within the confines of a Web browser or other application module.

XI. Supplementation of opinions

60. 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: 11/14/2011

Tony D. Givargis, Ph.D.

EXHIBIT A

Tony Givargis
University of California, Irvine
Computer Science
Irvine, CA 92697-3435
givargis@uci.edu
(949) 824-9357

Objective

Research and teaching in the areas of computer science with emphasis on embedded systems.

Education

- Ph.D., Computer Science, University of California, Riverside, 2001.
- B.S., Computer Science, University of California, Riverside, 1997.

Position

- Associate Dean for Student Affairs, School of Information & Computer Sciences, University of California, Irvine, 2011-present.
- Professor, Department of Computer Science, University of California, Irvine, 2011-present.
- Associate Professor, Department of Computer Science, University of California, Irvine, 2007-2011.
- Assistant Professor, Department of Computer Science, University of California, Irvine, 2001-2007.

Member

- Institute of Electrical and Electronics Engineers (IEEE).
- Association for Computing Machinery (ACM).
- Special Interest Group on Design Automation Executive Committee (SIGDA).
- Center for Embedded Computer Systems (CECS) at University of California, Irvine.

Service

Editorial Board

- Associate Editor, ACM Transactions on Embedded Computing Systems (TECS), 2008-present.
- Online Editor, Odysci, 2010-2011.
- Associate Editor, ACM SIGDA Bimonthly Newsletter, 2005-2007.
- Guest Editor, International Journal of Parallel Programming (IJPP), 2007.
- Guest Editor, International Journal of Parallel Programming (IJPP), 2006.
- Associate Editor, Journal of Embedded Computing (JEC), 2004-2006.

Distinguished Service

- Executive Committee Member, ACM Special Interest Group on Design Automation (SIGDA), 2009-2011.
- Technical Program Committee Chair, International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2010 (Co-Chair).
- Technical Program Committee Chair, IFIP Workshop on Software Technologies for Future Embedded & Ubiquitous Systems (SEUS), 2008 (Co-Chair).
- Technical Program Committee Chair, Special Interest Group on Design Automation (SIGDA) Ph.D. Forum at Design Automation Conference (DAC), 2007 (Chair).
- Technical Program Committee Chair, Special Interest Group on Design Automation (SIGDA) Technical Committee on System Design, 2006 (Co-Chair).
- Technical Program Committee Chair, Special Interest Group on Design Automation (SIGDA) Ph.D. Forum at Design Automation Conference (DAC), 2006 (Co-Chair).
- Special Sessions Chair, International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2005.

Technical Program Committee Member

- International Conference on Computer Aided Design (ICCAD), 2010-present.
- International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), 2009-present.
- International Symposium on Low Power Electronics and Design (ISLPED), 2007-present.
- International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2003-present.
- ACM Student Research Competition at DAC, 2010.
- Special Interest Group on Design Automation (SIGDA) Ph.D. Forum at Design Automation Conference (DAC), 2005-2008.
- International Conference on Embedded Software and Systems (ICESS), 2008.
- International Workshop on Embedded Single and Multicore Systems on Chips (MCSoC), 2007.
- International Workshop on Embedded Software Optimization (ESO), 2006.
- International Workshop on Logic and Synthesis (IWLS), 2004-2006.
- International Workshop on SoC and MCSoC Design (SoC), 2006.
- International Workshop on Embedded Computing (EC), 2006.
- International Conference on Embedded And Ubiquitous Computing (EUC), 2005.
- International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), 2005.
- International Conference on Computer Aided Design (ICCAD), 2003, 2004, 2005.
- Asia and South Pacific Design Automation Conference (ASP-DAC), 2003.
- International Workshop on Embedded System Codesign (ESCODES), 2002.

Topic Chair

- Embedded Software: International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2006.
- Micro-Architecture and Memory Optimizations: International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2005.
- System-Level Optimization: International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2004.

Session Chair

- International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2009.
- International Conference on Architecture, Compilers, and Synthesis for Embedded Systems (CASES), 2009.
- Design Automation Conference (DAC), 2009.
- Design Automation Conference (DAC), 2008.
- Design Automation and Test in Europe (DATE), 2008.
- Design Automation Conference (DAC), 2007.
- International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), 2005.
- International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2005.
- International Symposium on Low Power Electronics and Design (ISLPED), 2005.
- International Conference on Computer Aided Design (ICCAD), 2005.
- International Conference on Computer Aided Design (ICCAD), 2004.
- International Conference on Computer Aided Design (ICCAD), 2003.
- International Workshop on Hardware/Software Codesign (CODES), 2002.
- International Workshop on Embedded System Codesign (ESCODES), 2002.
- International Conference on Computer Aided Design (ICCAD), 2001.

Conference Organizer

- Finance Chair, Embedded Systems Week (ESWEEK), 2008.
- Finance Chair, International Symposium on Low Power Electronics and Design (ISLPED), 2007.
- Student Travel Grants Chair, International Conference on Supercomputing (ICS), 2006.
- Web Chair, International Symposium on Low Power Electronics and Design (ISLPED), 2006.
- Audio Visual Chair, International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2006.
- Finance Chair, Special Interest Group on Design Automation (SIGDA) Ph.D. Forum at Design Automation Conference (DAC), 2005.

- Publicity Chair, International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2004.
- Poster Committee Chair, Southern California Embedded Systems Symposium (SCESS), 2003.
- Local Chair and Treasurer, International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2003.
- Local Chair, International Workshop on Languages, Compilers, and Tools for Embedded Systems (LCTES), 2003.
- Finance and Registration Co-Chair, International Symposium on High Performance Computer Architecture (HPCA), 2003.

Reviewer & Panelist

- National Science Foundation (NSF).
- Kentucky Science & Engineering Foundation (KSEF).
- University of California Microelectronics Innovation and Computer Research Opportunities (UC-MICRO).
- Council of Physical Sciences of the Netherlands Organization for Scientific Research (NWO).
- IEEE Computer.
- IEEE Transactions on Computers (TC).
- IEEE Transactions on Computer Aided Design (TCAD).
- IEEE Transactions on Very Large Scale Integration Systems (TVLSI).
- IEEE Transactions on Design & Test of Computers (TD&T).
- IEEE Transactions on Circuits and Systems II (TCAS-II).
- ACM Transactions on Embedded Computing Systems (TECS).
- ACM Transactions on Design Automation of Electronic Systems (TODAES).
- IEE Proceedings - Computers and Digital Techniques (IEE-C&DT).
- Springer Design Automation for Embedded Systems (SDAES).
- Cluwer Design Automation for Embedded Systems (CDAES).
- Elsevier Journal of Microprocessors and Microsystems (MICPRO).
- Elsevier Journal of System Architecture (JSA).
- Asia and South Pacific Design Automation Conference (ASP-DAC).
- Design Automation Conference (DAC).
- International Conference on Computer Aided Design (ICCAD).
- International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS).
- International Symposium on Micro-architecture (MICRO).
- International Workshop on Compilers, Architecture, and Synthesis for Embedded Systems (CASES).
- International Workshop on Logic and Synthesis (IWLS).

University

- Member, University Committee on Computing and Communications (UCCC), 2007-2010.

Campus

- Member, Council on Research, Computing and Library Resources (CORCLR), 2007-2010.
- Chair, Faculty Board for Undecided/Undeclared Students: A Board of the Council on Educational Policy, 2003-2007.

School

- Associate Dean, ICS Student Affairs, 2011-present.
- Chair, ICS Strategic Planning: Graduate Education Group, 2011-2012.
- Member, Task Force on ICS First Year Curriculum, 2010-2011.
- Chair, Task Force on the CS Major, 2010-2011.
- Member, Computer Science Chair Recommendation Committee, 2009-2010.
- Chair, Computing & Network Policy, 2009-2010.
- Member, Computing & Network Policy, 2008-2009.
- Member, CS&E Steering Committee, 2003-2006.
- Member, DARPA Grand Challenge Team eXtreme Anteater Racers (XAR) Group, 2004-2005.
- Member, Entrepreneurship Committee, 2004-2005.
- Member, Executive Committee, 2003-2004.
- Member, Transition Committee, 2003-2004.
- Member, Faculty Recruit Committee (Ubiquitous Computing), 2003-2004.
- Member, Faculty Recruit Committee (Embedded Systems), 2002-2003.
- Member, CS Degree Program Committee, 2002-2003.
- Member, Graduate Recruit & Admissions Committee, 2002-2003.
- Member, CS&E Degree Program Committee, 2001-2002.

Awards

- NSF Grant (#1136146), National Science Foundation, $1.5M, 2011.
- ASEE ECE Division Hewlett-Packard Frederick Emmons Terman Award, 2011.
- NSF Grant (#1016789), National Science Foundation, $200K, 2010.
- ICS Dean's Award for Excellence in Undergraduate Teaching, University of California, Irvine, 2010.
- NSF Grant (#0837124), National Science Foundation, $70K, 2009.

- Best Paper, International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), 2008.
- Research & Travel Award, Council on Research, Computing, and Library Resources (CORCLR), University of California, Irvine, 2007.
- NSF Grant (#0749508), National Science Foundation, $200K, 2007.
- Faculty Desktop Computing Initiative Award, University of California, Irvine, 2007.
- SIGDA Technical Leadership Award, American Computing Machinery (ACM), 2007.
- Best Paper, American Control Conference (ACC), 2006.
- Best Paper, ACM Transactions on Design Automation of Electronic Systems (TODAES), 2006.
- Research & Travel Award, Council on Research, Computing, and Library Resources (CORCLR), University of California, Irvine, 2005.
- Collaborative Research Initiation Award (CRIA), School of Information and Computer Sciences, University of California, Irvine, 2005.
- Chancellor's Award for Excellence in Fostering Undergraduate Research, University of California, Irvine, 2005.
- Research & Travel Grant, School of Information and Computer Sciences, University of California, Irvine, 2004.
- Ted & Janice Smith Faculty Seed Funding Award, 2004.
- Discovery Grant, University of California, 2003.
- Research Grant, Microsoft Corporation & University of California MICRO Matching Funds, 2003.
- Excellence in Teaching Award, Instructional Resource Center/Division of Undergraduate Education, University of California, Irvine, 2003.
- Research & Travel Grant, School of Information and Computer Science, University of California, Irvine, 2003.
- Equipment Donation, Xilinx University Program, 2002.
- NSF ITR Grant, National Science Foundation, 2002.
- Research & Travel Grant, School of Information and Computer Sciences, University of California, Irvine, 2002.
- Equipment Donation, Xilinx University Program, 2001.
- Outstanding Ph.D. Thesis, Department of Computer Science & Engineering, University of California, Riverside, 2001.
- Best Paper, Design Automation and Test in Europe (DATE), 2000.
- GAANN Fellowship, Department of Computer Science & Engineering, University of California, Riverside, 1998.
- Graduate Scholarship, Design Automation Conference, 1998.
- Scholarship, International Council on Systems Engineering Inland Empire, 1997.

- MICRO Fellowship, Department of Computer Science & Engineering, University of California, Riverside, 1997.
- Outstanding Student Award, College of Engineering, University of California, Riverside, 1997.
- Outstanding Academic Program Excellence, Honors Convocations, University of California, Riverside, 1997.

Presentations

Tutorials

- ESL Design: Why the Time is Right and What are the Key Enabling Technologies. IEEE International Conference on Computer Aided Design (ICCAD) 2005.
- New Developments in Embedded System Design: Software for Embedded System. IEEE International Conference on Computer Design (ICCD) 2005.

Patents

Issued

P.9 A. Nacul, T. Givargis. Phantom Serializing Compiler and Method of Operation of Same. United States Patent, 7,886,283 , February 2011.

P8. J. Addink, S. Addink, T. Givargis. Methods and apparatus for using water use signatures and water pressure in improving water use efficiency. United States Patent 7,330,796, February 2008.

P7. J. Addink, S. Addink, T. Givargis. Methods and Apparatus for Using Water use Signatures in Improving Water use Efficiency. United States Patent 6,963,808, November 2005.

P6. J. Addink, T. Givargis. Interactive Irrigation System. United States Patent 6,950,728, September 2005.

P5. J. Addink, K. Buhler, T. Givargis. Modifying Irrigation Schedules of Existing Irrigation Controllers. United States Patent 6,892,114, May 2005.

P4. J. Henkel, T. Givargis, F. Vahid. Method for Core-Based System-Level Power Modeling using Object-Oriented Techniques. United States Patent 6,865,526, March 2005.

P3. K. Buhler, T. Givargis. Two Tire Irrigation Valve Controller. United States Patent 6,812,826, November 2004.

P2. J. Addink, T. Givargis. Detecting Weather Sensor Malfunctions. United States Patent 6,714,134, March 2004.

P1. J. Addink, K. Buhler, T. Givargis. Irrigation Accumulation Controller. United States Patent 6,298,285, October 2001.

Publications

Book

B3. F. Vahid, T. Givargis. Programming Embedded Systems - An Introduction to Time-Oriented Programming. UniWorld Publishing, July 2010. www.programmingembeddedsystems.com.

B2. A. Nacul, M. Lajolo, T. Givargis. Interface-Centric Abstraction level for Rapid Hardware/Software Integration, Book Chapter in Applications of Specification And Design Languages for SOCs. Springer, ISBN: 1-4020-4997-8, July 2006.

B1. F. Vahid, T. Givargis. Embedded System Design: A Unified Hardware/Software Introduction. John Wiley and Sons, ISBN: 0471386782, October 2001.

Journal

J20. C. Huang, F. Vahid, and T. Givargis. A Custom FPGA Processor for Physical Model Ordinary Differential Equation Solving. IEEE Embedded Systems Letters, to appear.

J19. S. Choudhuri, T. Givargis. Deterministic Service Guarantees for NAND Flash using Partial Block Cleaning. Academy Publisher Journal of Software (JSW), vol. 4, no. 7, pp. 728-737, September 2009.

J18. M.A. Ghodrat, T. Givargis, A. Nicolau. Optimizing Control Flow in Loops using Interval and Dependence Analysis. Springer Journal on Design Automation of Embedded Systems (DAES), vol. 13, no. 3, pp. 193-221, September 2009.

J17. S. Sirowy, D. Sheldon, T. Givargis, F. Vahid. Virtual Microcontrollers. ACM SIGBED Review, vol. 6, no. 1, January 2009.

J16. A. Nacul, T. Givargis. Synthesis of Time-Constrained Multitasking Embedded Software. ACM Transactions on Design Automation of Electronic Systems (TODAES), vol. 11, no. 4, pp. 822-847, October 2006.

J15. M.A. Ghodrat, T. Givargis, A. Nicolau. Expression Equivalence Checking using Interval Analysis. IEEE Transactions on Very Large Scale Integration Systems (TVLSI), vol. 14, no. 8, pp. 830-842, August 2006.

J14. C.V. Lopes, A. Haghighat, A. Mandal, T. Givargis, P. Baldi. Localization of Off-the-Shelf Mobile Devices Using Audible Sound: Architectures, Protocols and Performance Assessment. ACM Mobile Computing and Communications Review (MC2R), vol. 10, no. 2, pp. 38-50, April 2006.

J13. T. Givargis. Zero Cost Indexing for Improved Processor Cache Performance. ACM Transactions on Design Automation of Electronic Systems (TODAES), vol. 11, no. 1, pp. 3-25, January 2006. *Received the 2006 TODAES Best Paper Award.*

J12. T. Givargis, David Eppstein. Memory Reference Caching for Activity Reduction on Address Buses. Elsevier Journal of Microprocessors and Microsystems (MICPRO), vol. 29, no. 4, pp. 145-153, May 2005.

J11. A. Ghosh, T. Givargis. Cache Optimization for Embedded Processor Cores: An Analytical Approach. ACM Transactions on Design Automation of Electronic Systems (TODAES), vol. 9, no. 4, pp. 419-440, October 2004.

J10. A. Nacul, T. Givargis. Adaptive Cache Management for Low Power Embedded Systems. Korea Multimedia Society, Key Technology of Next Generation IT, ISSN 1229-778X, pp. 30-39, December 2003.

J9. T. Givargis, F. Vahid, J. Henkel. Instruction-Based System-level Power Evaluation of System-on-a-Chip Peripheral Cores. IEEE Transactions on Very Large Scale Integration Systems (TVLSI), vol. 10, no. 6, pp. 856-863, December 2002.

J8. T. Givargis, F. Vahid, J. Henkel. System-Level Exploration for Pareto-Optimal Configurations in Parameterized System-on-a-Chip. IEEE Transactions on Very Large Scale Integration Systems (TVLSI), vol. 10, no. 4, pp. 416-422, December 2002.

J7. T. Givargis, F. Vahid. Platune: A Tuning Framework for System-on-a-Chip Platforms. IEEE Transactions on Computer Aided Design (TCAD), vol. 21, no. 11, pp. 1317-1327, November 2002.

J6. F. Vahid, T. Givargis, S. Cotterell. Power Estimator Development for Embedded System Memory Tuning. Journal of Circuits, Systems, and Computers (JCSC), vol. 11, no. 5, pp. 459-476, October 2002.

J5. T. Givargis, F. Vahid. Tuning of Cache Ways and Voltage for Low-Energy Embedded System Platforms. Springer Journal on Design Automation of Embedded Systems, vol. 7, issue 1-2, pp. 35-51, September 2002.

J4. T. Givargis, F. Vahid, J. Henkel. Evaluating Power Consumption of Parameterized Cache and Bus Architectures in System-on-a-Chip Designs. IEEE Transactions on Very Large Scale Integration Systems (TVLSI), vol. 9, no. 4, pp. 500-508, August 2001.

J3. F. Vahid, T. Givargis. Platform Tuning for Embedded Systems Design. IEEE Computer, vol. 34, no. 3, pp. 112-114, March 2001.

J2. J. Farrell, T. Givargis, M. Barth. Real-Time Differential Carrier Phase GPS-Aided INS. IEEE Transactions on Control Systems Technology (TCST), vol. 8, no. 4, pp. 709-721, July 2000.

J1. J. Farrell, T. Givargis. Differential GPS Reference Station Algorithm - Design and Analysis. IEEE Transactions on Control Systems Technology (TCST), vol. 8, no. 3, pp. 519-531, May 2000.

Conference

C47. B. Miller, F. Vahid, T. Givargis. Demonstration of Digital Mockups for the Testing of a Medical Ventilator. ACM SIGHIT International Health Informatics Symposium (IHIS), to appear.

C46. B. Miller, F. Vahid, T. Givargis. Application-Specific Codesign Platform Generation for Digital Mockups in Cyber-Physical Systems. Electronic System Level Synthesis Conference (ESLsyn), to appear.

C45. M.A. Ghodrat, T. Givargis. Efficient Dynamic Voltage/Frequency Scaling through Algorithmic Loop Transformation. International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), pp. 203-209, Grenoble, October 2009.

C44. S. Sirowy, F. Vahid, T. Givargis. Digitally-Bypassed Transducers: Interfacing Digital Mockups to Real-Time Medical Equipment. International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS), pp. 919-922, Minneapolis, September 2009.

C43. A. Ghosh, T. Givargis. Source Routing made Practical in Embedded Networks. International Conference on Computer Communications and Networks (ICCCN), pp. 1-6, San Francisco, August 2009.

C42. A. Ghosh, T. Givargis. QoS Routing in Wired Sensor Networks with Partial Updates. World Academy of Science, Engineering and Technology (WASED), pp. 389-393, Oslo, July 2009.

C41. S.K. Mylavarapu, S. Choudhuri, A. Shrivastava, J. Lee, T. Givargis. FSAF: File System Aware Flash Translation Layer for NAND Flash Memories. Design Automation and Test in Europe (DATE), pp. 339-344, Dresden, April 2009.

C40. S. Choudhuri, T. Givargis. FlashBox: A system for logging non-deterministic events in deployed embedded systems. International ACM Symposium on Applied Computing (SAC), pp. 1676-1682, Honolulu, March 2009.

C39. M.A. Ghodrat, T. Givargis, A. Nicolau. Control Flow Optimization in Loops using Interval Analysis. International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), pp. 157-166, Atlanta, October 2008. *Received the 2008 CASES Best Paper Award.*

C38. F. Vahid, T. Givargis. Timing is Everything - Embedded Systems Demand Early Teaching of Structured Time-Oriented Programming. Workshop on Embedded Systems Education (WESE), Atlanta, October 2008.

C37. S. Sirowy, D. Sheldon, T. Givargis, F. Vahid. Virtual Microcontrollers. Workshop on Embedded Systems Education (WESE), Atlanta, October 2008.

C36. F. Vahid, T. Givargis. Highly-Cited Ideas in System Codesign and Synthesis. International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), pp. 191-196, Atlanta, October 2008.

C35. S. Choudhuri, T. Givargis. Deterministic Service Guarantees for NAND Flash using Partial Block Cleaning. International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), pp. 19-24, Atlanta, October 2008.

C34. S. Choudhuri, T. Givargis. Real-Time Access Guarantees for NAND Flash using Partial Block Cleaning. Workshop on Software Technologies for Future Embedded & Ubiquitous Systems (SEUS), pp. 138-149, Italy, September 2008.

C33. A. Ghosh, T. Givargis. A Software Architecture for Accessing Data in Sensor Networks. International Conference on Networked Sensing Systems (INSS), pp. 67-70, Japan, June 2008.

C32. S. Choudhuri, T. Givargis. Performance Improvement of Block Based NAND Flash Translation Layer. International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS), pp. 257-262, Salzburg, September 2007.

C31. M.A. Ghodrat, T. Givargis., A. Nicolau. Short-Circuit Compiler Transformation: Optimizing Conditional Blocks. Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 504-510, Tokyo, January 2007.

C30. S. Choudhuri, T. Givargis. System Architecture for Software Peripherals. Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 56-61, Tokyo, January 2007.

C29. A. Nacul, T. Givargis. Phantom: A Serializing Compiler for Multitasking Embedded Software. American Control Conference (ACC), pp. 1918-1923, Minneapolis, June 2006. *Received the 2006 ACC Best Paper Award.*

C28. M.A. Ghodrat, T. Givargis, A. Nicolau. Equivalence Checking of Arithmetic Expressions using Fast Evaluation. International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), pp. 147-156, San Francisco, September 2005.

C27. A. Nacul, T. Givargis. Lightweight Multitasking Support for Embedded Systems using the Phantom Serializing Compiler. Design Automation and Test in Europe (DATE), pp. 742-747, Munich, March 2005.

C26. A. Ghosh, T. Givargis. LORD: A Localized, Reactive and Distributed Protocol for Node Scheduling in Wireless Sensor Networks. Design Automation and Test in Europe (DATE), pp. 190-195, Munich, March 2005.

C25. A. Mandal, C.V. Lopes, T. Givargis, A. Haghighat, R. Jurdak, P. Baldi. Beep: 3D Indoor Positioning Using Audible Sound. IEEE Consumer Communications and Networking Conference (CCNC), pp. 348-353, Las Vegas, January 2005.

C24. A. Nacul, T. Givargis. Code Partitioning for Synthesis of Embedded Applications with Phantom. International Conference on Computer-Aided Design (ICCAD), pp. 190-196, San Jose, November 2004.

C23. A. Nacul, T. Givargis. Dynamic Voltage and Cache Reconfiguration for Low Power. Design Automation and Test in Europe (DATE), pp. 1376-1377, Paris, February 2004.

C22. M. Buss, T. Givargis, N. Dutt. Exploring Efficient Operating Points for Voltage Scaled Embedded Processor Cores. Real-Time Systems Symposium (RTSS), pp. 275-281, Cancun, December 2003.

C21. A. Ghosh, T. Givargis. Cache Optimization for Embedded Processor Cores: An Analytical Approach. International Conference on Computer-Aided Design (ICCAD), pp. 342-347, San Jose, November 2003.

C20. T. Givargis. Improved Indexing for Cache Miss Reduction in Embedded Systems. Design Automation Conference (DAC), pp. 872-880, Anaheim, June 2003.

C19. A. Ghosh, T. Givargis. Analytical Design Space Exploration of Caches for Embedded Systems. Design Automation and Test in Europe (DATE), pp. 650-655, Munich, March 2003.

C18. T. Givargis, D. Eppstein. Reference Caching Using Unit Distance Redundant Codes for Activity Reduction on Address Buses. International Workshop on Embedded System Hardware/Software Codesign (ESCODES), San Jose, September 2002.

C17. M. Palesi, T. Givargis. Multi-Objective Design Space Exploration Using Genetic Algorithms. International Workshop on Hardware/Software Codesign (CODES), Estes Park, May 2002.

C16. T. Givargis, F. Vahid, J. Henkel. System-Level Exploration for Pareto-Optimal Configurations in Parameterized Systems-on-a-Chip. International Conference on Computer-Aided Design (ICCAD), San Jose, November 2001.

C15. T. Givargis, F. Vahid. J. Henkel. Trace-Driven System-Level Power Evaluation of System-on-a-Chip Peripheral Cores. Asia and South Pacific Design Automation Conference (ASP-DAC), Yokohama, January 2001.

C14. G. Stitt, F. Vahid, T. Givargis, R. Lysecky. A First-Step Towards an Architecture Tuning Methodology. International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES), San Jose, November 2000.

C13. T. Givargis, F. Vahid, J. Henkel. Instruction-Based System-Level Power Evaluation of System-on-a-Chip Peripheral Cores. International Symposium on System Synthesis (ISSS), Madrid, September 2000.

C12. R. Lysecky, F. Vahid, T. Givargis. Experiments with the Peripheral Virtual Component Interface. International Symposium on System Synthesis (ISSS), Madrid, September 2000.

C11. T. Givargis, F. Vahid. Parameterized System Design. International Workshop on Hardware/Software Codesign (CODES), San Diego, May 2000.

C10. T. Givargis, F. Vahid, J. Henkel. Fast Cache and Bus Power Estimation for Parameterized System-on-a-Chip Design. Design Automation and Test in Europe (DATE), Paris, March 2000.

C9. R. Lysecky, F. Vahid, T. Givargis. Techniques for Reducing Read Latency of Core Bus Wrappers. Design Automation and Test in Europe (DATE), Paris, March 2000. *Received the 2000 DATE Best Paper Award.*

C8. T. Givargis, F. Vahid. J. Henkel. A Hybrid Approach for Core-Based System-Level Power Modeling. Asia and South Pacific Design Automation Conference (ASPDAC), Yokohama, January 2000.

C7. T. Givargis, J. Henkel, F. Vahid. Interface and Cache Power Exploration for Core- Based Embedded System Design. International Conference on Computer-Aided Design (ICCAD), San Jose, November 1999.

C6. R. Lysecky, F. Vahid, T. Givargis, R. Patel. Pre-Fetching for Improved Core Interfacing. International Symposium on System Synthesis (ISSS), San Jose, November 1999.

C5. J. Farrell, T. Givargis. Experimental Differential GPS Reference Station Evaluation. American Control Conference (ACC), San Diego, June 1999.

C4. J. Farrell, T. Givargis. M. Barth. Differential Carrier Phase GPS-Aided INS for Automotive Applications. American Control Conference (ACC), San Diego, June 1999.

C3. F. Vahid, T. Givargis. The Case for a Configure-and-Execute Paradigm. International Workshop on Hardware/Software Codesign (CODES), Rome, May 1999.

C2. F. Vahid, T. Givargis. Incorporating Cores into System-Level Specification. International Symposium on System Synthesis (ISSS), Hsinchu, December 1998.

C1. T. Givargis, F. Vahid. Interface Exploration for Reduced Power in Core-Based Systems. International Symposium on System Synthesis (ISSS), Hsinchu, December 1998.

Workshop

W2. A. Nacul, M. Lajolo, T. Givargis. Interface-Centric Abstraction Level for Rapid Hardware/Software Integration. Forum on Specification and Design Languages (FDL), Lausanne, September 2005.

W1. A. Haghighat, C. Lopes, T. Givargis, and A. Mandal. Location-Aware Web System. Workshop on Building Software for Pervasive Computing at the Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) Conference, Vancouver, October 2004.

Miscellaneous

M1. U. Brinkschulte, M. Cinque, T. Givargis, S. Russo. Guest Editorial. Journal of Software, vol. 4, no. 7, pp. 631-633, Septempber 2009.

Affiliated Students

Ph.D. Final Defense Committee Chair

- Mohammad Ali Ghodrat, Department of Computer Science, University of California, Irvine, 9/1/2009.
- Siddharth Choudhuri, Department of Computer Science, University of California, Irvine, 1/5/2009.
- Arijit Ghosh, Department of Computer Science, University of California, Irvine, 07/09/2008.
- Andre Nacul, Department of Computer Science, University of California, Irvine, 04/23/2007.

Ph.D. Final Defense Committee Member

- Michael Bebenita, Department of Computer Science, University of California, Irvine, 10/11/2011.
- Jinsik Kim, Department of Electrical Engineering & Computer Science, University of California, Irvine, 05/18/2010.
- Yonghyun Hwang, Department of Computer Science, University of California, Irvine, 12/16/2009.
- Babak Salamat, Department of Computer Science, University of California, Irvine, 06/8/2009.
- Aseem Gupta, Department of Electrical Engineering & Computer Science, University of California, Irvine, 05/29/2009.
- Gabor Madl, Department of Computer Science, University of California, Irvine, 05/27/2009.
- Love Singhal, Department of Computer Science, University of California, Irvine, 01/13/2009.
- Lei Zhou, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/03/2008.
- Seung-Eun Lee, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/03/2008.
- Minyoung Kim, Department of Computer Science, University of California, Irvine, 7/8/2008.
- Jun Ho Bahn, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/20/2007.
- Daniel Jesus Valencia Sanchez, Department of Computer Science, University of California, Irvine, 09/10/2007.
- Chen Liu, Department of Electrical Engineering & Computer Science, University of California, Irvine, 08/03/07.
- Vasanth Venkatachalam, Department of Computer Science, University of California, Irvine, 5/14/2007.
- Ning Wang, Department of Computer Science, University of California, Irvine, 5/14/2007.
- Kiran Ramineni, Department of Computer Science, University of California, Irvine, 3/13/2007.
- Shireesh Verma, Department of Computer Science, University of California, Irvine, 2/26/2007.
- Ilya Issenin, Department of Computer Science, University of California, Irvine, 12/06/2006.
- Hooman Parizi, Department of Electrical Engineering & Computer Science, University of California, Irvine, 11/14/2006.
- Partha Biswas, Department of Computer Science, University of California, Irvine, 03/02/2006.
- Samar Abdi, Department of Computer Science, University of California, Irvine, 11/28/2005.
- Chengzhi Pan, Department of Electrical Engineering & Computer Science, University of California, Irvine, 09/07/2005.
- Haobo Yu, Department of Computer Science, University of California, Irvine, 11/30/2004.
- Shean McMahon, Department of Computer Science, University of California, Irvine, 11/01/2004.
- Mahesh Naga Mamidipaka, Department of Computer Science, University of California, Irvine, 08/23/2004.
- Dongwan Shin, Department of Computer Science, University of California, Irvine, 04/16/2004.
- Junyu Peng, Department of Computer Science, University of California, Irvine, 04/02/2004.

Ph.D. Qualifying Exam Committee Chair

- Siddharth Choudhuri, Department of Computer Science, University of California, Irvine, 12/21/2006.
- Mohammad Ali Ghodrat, Department of Computer Science, University of California, Irvine, 03/25/2005.
- Arijit Ghosh, Department of Computer Science, University of California, Irvine, 09/14/2004.
- Andre Nacul, Department of Computer Science, University of California, Irvine, 05/26/2004.

Ph.D. Qualifying Exam Committee Member

- Mehryar Rahmatian, Department of Computer Science, University of California, Irvine, 9/16/2010.
- Deepak Mishra, Department of Computer Science, University of California, Irvine, 9/15/2010.
- Dali Zhao, Department of Computer Science, University of California, Irvine, 9/2/2010.
- Tae Su Kim, Department of Computer Science, University of California, Irvine, 3/19/2010.
- Zhiming Chen, Department of Computer Science & Engineering, University of California, Riverside, 12/10/2009.
- Hessam Kooti, Department of Computer Science, University of California, Irvine, 12/7/2009.
- Vahid Salmani, Department of Computer Science & Engineering, University of California, Riverside, 12/2/2009.
- Marcelo Cintra, Department of Computer Science, University of California, Irvine, 11/17/2009.
- Patricia Lee, Department of Computer Science, University of California, Irvine, 9/18/2009.
- Rosario Cammarota, Department of Computer Science, University of California, Irvine, 9/17/2009.
- Kazuyuki Tanimura, Department of Computer Science, University of California, Irvine, 9/3/2009.
- Matthew Badin, Department of Computer Science, University of California, Irvine, 6/4/2009.
- Michael Bebenita, Department of Computer Science, University of California, Irvine, 5/29/2009.
- Mason Chang, Department of Computer Science, University of California, Irvine, 5/27/2009.
- Sangwon Chae, Department of Computer Science, University of California, Irvine, 12/09/2008.
- Wendy Zhang, Department of Computer Science, University of California, Irvine, 12/08/2008.
- Fred Tzeng, Department of Computer Science & Engineering, University of California, Riverside, 1/31/2008.
- Minyoung Kim, Department of Computer Science, University of California, Irvine, 01/15/2008.
- AmirHossein GholamiPour, Department of Computer Science, University of California, Irvine, 12/11/2007.
- Shahin Golshan, Department of Computer Science, University of California, Irvine, 12/10/2007.
- Scott Hendrickson, Department of Informatics, University of California, Irvine, 11/16/2007.
- David Sheldon, Department of Computer Science & Engineering, University of California, Riverside, 10/24/2007.

- Scott Sirowy, Department of Computer Science & Engineering, University of California, Riverside, 10/24/2007.
- Yonghyun Hwang, Department of Computer Science, University of California, Irvine, 09/19/2007.
- Zhen Zhang, Department of Computer Science, University of California, Irvine, 09/4/2007.
- Jing Qian, Department of Electrical Engineering & Computer Science, University of California, Irvine, 09/4/2007.
- Babak Salamat, Department of Computer Science, University of California, Irvine, 05/15/2007.
- Sudeep Pasricha, Department of Computer Science, University of California, Irvine, 02/07/2007.
- Deyi Pi, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/5/2006.
- Seung Eun Lee, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/1/2006.
- Jayram Nageswaran, Department of Computer Science, University of California, Irvine, 11/14/2006.
- Gabor Madl, Department of Computer Science, University of California, Irvine, 09/06/2006.
- Lei Zhou, Department of Electrical Engineering & Computer Science, University of California, Irvine, 06/16/2006.
- Love Singhal, Department of Computer Science, University of California, Irvine, 06/12/2006.
- Ilya Issenin, Department of Computer Science, University of California, Irvine, 05/23/2006.
- Sevin Fide, Department of Electrical Engineering & Computer Science, University of California, Irvine, 05/16/2006.
- Farzad Etemadi, Department of Electrical Engineering & Computer Science, University of California, Irvine, 04/13/2006.
- Vasanth Venkatachalam, Department of Computer Science, University of California, Irvine, 04/11/2006.
- Jiwon Hahn, Department of Electrical Engineering & Computer Science, University of California, Irvine, 03/28/2006.
- Ersin Sengul, Department of Electrical Engineering & Computer Science, University of California, Irvine, 03/08/2006.
- Daniel Jesus Valencia Sanchez, Department of Computer Science, University of California, Irvine, 02/24/2006.
- Hooman Torabi Parizi, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/15/2005.
- Afshin Niktash, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/13/2005.
- Jun Ho Bahn, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/7/2005.
- Chulsung Park, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/6/2005.

- Aseem Gupta, Department of Electrical Engineering & Computer Science, University of California, Irvine, 09/14/2005.
- Gunar Schirner, Department of Computer Science, University of California, Irvine, 09/06/2005.
- Pramod Chandraiah, Department of Electrical Engineering & Computer Science, University of California, Irvine, 09/06/2005.
- Yan Huang, Department of Computer Science, University of California, Irvine, 08/24/2005.
- Partha Biswas, Department of Computer Science, University of California, Irvine, 06/13/2005.
- Jiming Liu, Department of Computer Science, University of California, Irvine, 05/06/2005.
- Pablo Diaz Gutierrez, Department of Computer Science, University of California, Irvine, 03/24/2005.
- Srinivas Vadlamani, Department of Electrical Engineering & Computer Science, University of California, Irvine, 03/16/2005.
- Rafael Lopez, Department of Computer Science, University of California, Irvine, 02/02/2005.
- Kiran Ramineni, Department of Computer Science, University of California, Irvine, 12/09/2004.
- Seung-mok Yoo, Department of Electrical Engineering & Computer Science, University of California, Irvine, 12/07/2004.
- Chen Liu, Department of Electrical Engineering & Computer Science, University of California, Irvine, 11/30/2004.
- Ashish Bhargave, Department of Electrical Engineering & Computer Science, University of California, Irvine, 11/2005/2004.
- Cristian Petrescu-Prahova, Department of Computer Science, University of California, Irvine, 09/13/2004.
- Shireesh Verma, Department of Computer Science, University of California, Irvine, 09/09/2004.
- Chris Fensch, Department of Computer Science, University of California, Irvine, 09/08/2004.
- Ning Wang, Department of Computer Science, University of California, Irvine, 09/01/2004.
- Enis Akay, Department of Electrical Engineering & Computer Science, University of California, Irvine, 06/2004/2004.
- Amir Kamalizad, Department of Electrical Engineering & Computer Science, University of California, Irvine, 06/03/2004.
- Deepak Chandra, Department of Computer Science, University of California, Irvine, 03/30/2004.
- Chunhui Zhang, Department of Electrical Engineering & Computer Science, University of California, Irvine, 03/15/2004.
- David Lee, Department of Chemistry, University of California, Irvine, 05/07/2003.
- Jinfeng Liu, Department of Electrical Engineering & Computer Science, University of California, Irvine, 03/20/2003.
- Mehrdad Reshadi, Department of Computer Science, University of California, Irvine, 03/19/2003.
- Chengzhi Pan, Department of Electrical Engineering & Computer Science, University of California, Irvine, 03/12/2003.

- Samar Abdi, Department of Computer Science, University of California, Irvine, 09/12/2002.
- Marcio Dias, Department of Informatics, University of California, Irvine, 06/03/2002.
- Chris Lüer, Department of Informatics, University of California, Irvine, 05/14/2002.
- Dongwan Shin, Department of Computer Science, University of California, Irvine, 09/20/2001.
- John Xie, Department of Computer Science, University of California, Irvine, 09/20/2001.
- Haobo Yu, Department of Computer Science, University of California, Irvine, 09/20/2001.
- Junyu Peng, Department of Computer Science, University of California, Irvine, 09/14/2001.

Undergraduate Research Opportunities Program (UROP) Faculty Mentor

- Manish Sinha, Information & Computer Sciences, University of California, Irvine, 2007.
- Hussien Sleiman, Information & Computer Sciences, University of California, Irvine, 2005.
- Luis Angel, Information & Computer Sciences, University of California, Irvine, 2005.
- Martin Yasin, Information & Computer Sciences, University of California, Irvine, 2005.
- Anton Popov, Information & Computer Sciences, University of California, Irvine, 2005.
- Nick Mangano, Information & Computer Sciences, University of California, Irvine, 2005.
- Harvey Herela, Information & Computer Sciences, University of California, Irvine, 2005.
- Chad Christensen, Information & Computer Sciences, University of California, Irvine, 2005.
- Adam Wasserstrom, Information & Computer Sciences, University of California, Irvine, 2005.
- Lu Q Zheng, Information & Computer Sciences, University of California, Irvine, 2005.
- Ray Shen, Information & Computer Sciences, University of California, Irvine, 2005.
- Long Ting Kan, Information & Computer Sciences, University of California, Irvine, 2005.

Summer Undergraduate Research Program (SURP) Faculty Mentor

- John Liu, Computer Science & Engineering, University of California, Riverside, 2003.
- Kenny Vu, Information & Computer Sciences, University of California, Irvine, 2002.