# Why the Proposed Article 2 Revisions[1] Fall Short For Embedded Systems

(Draft: November 18, 2000)[2]

Philip Koopman[3]
Assistant Professor
Electrical & Computer Engineering Department
Carnegie Mellon University
and
Cem Kaner[4]
Professor of Computer Sciences
Florida Institute of Technology

## Abstract

The new scope proposal for Article 2 attempts to solve the embedded software problem by declaring that an "integrated product" fits within the scope of Article 2 but that all other software does not. Unfortunately, this definition's view of embedded software is outdated. A manufacturer can easily work around it, designing embedded software that is excluded from Article 2's coverage.

## 1. Introduction

The new scope proposal rests on the definition of an integrated product:

> *"Integrated product" means goods that incorporate a copy of a computer program of a type that is not separately available as part of the transaction for the goods. The term does not include a computer, computer peripheral, or medium of fixation.*

The proposal comments on this definition as follows:

> *Many goods incorporate copies of computer programs that control various aspects of the performance of the goods. If the computer program is of a type that is not separately available as part of the transaction for the goods, the goods and the computer program together constitute an integrated product. Not every copy of a computer program incorporated into goods is an integrated product. If the buyer has a meaningful choice among programs, the copy of the program is not part of an integrated product.*
>
> *For example, compare (a) a sewing machine on which the stitching is controlled by an electronic controller incorporating a computer program with (b) a programmable calculator. In the case of the sewing machine, assume that the computer program that directs the actions of the controller is not available separately from the goods. In the case of the calculator, assume that either (i) the calculator can be purchased with a copy of a computer program already incorporated into it that enables complex financial calculations, or (ii) the calculator can be purchased without a copy of a financial-*

*calculations program in it and the program can be acquired separately as part of the transaction. The sewing machine and the copy of its program comprise an integrated product, while the calculator and the copy of its program do not comprise an integrated product. Note that if the sewing machine is available in two different versions – one that incorporates a computer program enabling 16 stitches and the other with a program enabling 32 stitches – but neither program is of a type that is available separately from the goods as part of the transaction, each version of the sewing machine constitutes an integrated product.*

*The "separately available" test applies only to the transaction at issue, and it applies only to the computer program itself. If the type of program at issue is not separately available as part of the transaction, the goods are an integrated product even if alternative programs are generally available in the marketplace. Similarly, if the seller gives the buyer a choice of parts that can be purchased separately from the goods and each part is identical except that it incorporates a copy of a different computer program, and if no other program is separately available for either part, the product that results from the combination of the goods and the part selected by the buyer is an integrated product. By contrast, if the program in the part that is selected by the buyer is of a type that is separately available, it is not part of an integrated product.*

*Although they may contain copies of computer programs, a computer, a computer peripheral, or a medium of fixation is not an integrated product. For example, if a buyer purchases a computer containing an operating system the computer is goods but the combination of the goods and the operating system is not an integrated product. This transaction is governed by Section 2-103(c), which adopts a gravamen of the action test pursuant to which Article 2 applies only to the goods. The same rule applies to a computer peripheral and to a medium of fixation.*

*The term "computer" is not defined in this Article. The lack of a definition should not prove troublesome as courts routinely use common sense when applying the definitions found in other statutes. For example, unauthorized access to a computer is a crime under various state statutes, but while the definition of computer in those statutes is broad, courts exercise common sense in applying the definition. Thus, while an automobile might contain a computer or several computers, the automobile itself is not a computer and breaking into it is not viewed as unauthorized access to a computer.*

Under proposed Section 2-103:

*(b) In applying the provisions of this article to an integrated product, the goods and any copy of an incorporated computer program are to be considered as a whole.*

*(c) Subject to subsection (b) as it applies to an integrated product, in a transaction that includes goods and information, including a copy of a computer program that is not incorporated in an integrated product, this article applies to the goods.*

Let's consider the sewing machine example in more detail, to illustrate one of the problems. The hypothetical sewing machine comes in two models, a 16-stitch version (with 16-stitch software) and a 32-stitch version (with 32-stitch software). A manufacturer *could* market the machine this way, but under the integrated product rule, the manufacturer would probably do it differently.

Suppose instead that the manufacturer sells a single sewing machine that comes by default with 4-bit stitching. The customer can upgrade to 16- or 32-stitch capability (and most customers will) either at time of sale or later. The upgrade software is stored on a disk and downloaded into the sewing machine's memory. While only one program (16- or 32-stitch) can be active at a time, the customer can buy a copy of both programs. In this case, the program is available separately and under the proposed Section 2-103, it will therefore be excluded from coverage under Article 2.

There is nothing special about sewing machines. The same problems will arise with software that controls cars, home heating systems, telephones, sound systems, home medical devices, and many other consumer products that would normally be classed as goods.

# 2. Definitional Problems

Another key problem is the exclusion of computers and computer peripherals from the category of integrated products. Proposed Article 2 does not define "computer" or "computer peripheral" but the software that is embedded in them or sold with them is not "integrated" and is to fall outside of the new scope of Article 2.

### *2.1 Definition of "computer"*

The word "computer" is clearly and unambiguously defined in UCITA's[5] section 102(a)9 to encompass virtually all digital computers, embedded or otherwise.

IEEE 610.10-1994[6] defines computer as

> "a device that consists of one or more processing units and peripheral units, that is controlled by internally stored programs, and that can perform substantial computations, including numerous arithmetic operations, or logic operations, without human intervention during a run. Note: may stand alone, or may consist of several interconnected units."

This definition is generally comparable to the UCITA definition in scope. It is worth noting that IEEE 610.10-1994 sees fit to identify an "embedded computer" as a subset of the class of all computers:

> "a computer system that is part of a larger system and which performs some of the requirements of that system; for example a computer system used in an aircraft or rapid transit system."

Thus, using either the UCITA definition of "computer" or the IEEE definition of "computer," the concept of a computer means all computers, not just desktop computers.

Here's an example of a computer:

> "Our Turbo City Corvette '82 - '84 Crossfire Upgrade Kit gives you: a 90's computer to make the Crossfire Fuel Injection react more quickly and more accurately. The package includes the 90's Crossfire upgrade computer with Eprom, crimping pliers, connectors and instructions to change-out the '82-'84 ECM. This will improve low and midrange performance, and fuel and spark delivery. The ECM will have our custom stock Crossfire chip. Once your upgrade computer is installed, we can provide special programming for replacement chips that will compensate for almost any type of performance modification or added power accessory. Any number of custom performance or special requirement chips will be made for your request, for only $129.00 +S&H per chip."[7]

This is being sold by a third party, not by the makers of the Corvette. Other discussions on the site make it clear that the software comes with a license. We believe that this transaction would be within the scope of UCITA and outside the scope of Article 2. The chip is merely a medium on which the software is stored.

Note that this chip is an EPROM (erasable programmable read only memory). Once you install the chip in the car, you can probably upgrade the software without changing the chip. Surely, that software upgrade is software like a word processing program is software.

Cars can come from the manufacturer with software that can be upgraded merely by downloading new software into the chips rather than swapping out chips. On what rational basis can we insist

that the original equipment manufacturer's software is integrated whereas a third party's software that performs the same functions is not?

It is very likely that all future automobile engine controllers will have reprogrammable memory (*e.g.*, flash memory) to reduce the cost of recalls in the event of a software defect being discovered, and it is common to have it even today. Currently flash memory is being used by cell phones, hard drives, network hubs, and at least some automotive engine controllers. (source: http://www.bizjournals.com/sanjose/stories/1996/09/23/story6.html, September 20, 1996 article on flash memory usage.)

The sewing machine example's vision of embedded software as being permanently and unalterably loaded on a chip (so that we buy different hardware to gain access to different software) is no longer representative of embedded software in the real world.

### *2.2 Definition of "computer peripheral"*

The phrase "computer peripheral" is not defined by UCITA or Article 2. There are two possible ways to clarify the meaning, both of which are too imperfect to be practical: definition by listing items, and definition by connectivity.

An attempt to list computer peripherals has the advantage of superficial clarity. A nice tidy list such as: "printer, scanner, keyboard, hard disk drive, floppy disk drive, CD-ROM drive, DVD-ROM drive, mouse, or trackball" is reasonably specific. However, it suffers in that new peripherals are continually being introduced (e.g., joystick, parallel-port video camera, virtual reality glove). Furthermore, even a seemingly obvious list and complete list would suffer problems discussed below and in the section on "gray area" devices.

The only plausible default definition of "computer peripheral" other than a list of specific items is "anything that is attached to a computer". IEEE standard 610.10-1994 defines a peripheral as:

> "...a device that operates in combination or conjunction with the computer but is not physically part of the computer and is not essential to the basic operation of the system; for example, printers, keyboards, graphic digital converters, disks, and tape drives."

This definition conveys the same general sense and has the virtue of being internationally standard technical terminology. But this approach is fraught with problems.

Let us take the example of an ordinary laser printer, which most people would agree is a computer peripheral when placed in a normal office setting. This printer could be connected via a cable directly to a computer, making it a classical peripheral device. Or it could be attached to a local area network. Does being attached via a network make a printer not a peripheral? Most would probably say not, since it does not change the inherent property of being a printer (especially models that are factory configured to work either via network or via dedicated peripheral cable right out of the box). But, does that make anything connected to a computer via a network or other indirect connection a peripheral? If so, then very soon a dizzying array of items could become computer peripherals merely by adding a network connection to an existing class of product that most clearly is not a computer peripheral, including:

- Internet-enabled household appliances ("smart" refrigerators, ovens, and so on)
- Gas and electric meters incorporating modems to dial in meter readings (these are already widely installed)
- Digital television recording devices that use a modem or cable modem to download new programming information
- Sensors that use embedded web servers to display status information (or perhaps in that case a household thermostat would really a "computer" instead of a "peripheral" – it will

be difficult to really know until each and every such type of device is dealt with by litigation).

If connectivity or the ability to send or receive information to a general purpose computer is used as the defining quality of being a computer peripheral, then a number of devices that are usually not considered to be peripherals would then be transformed, by fiat, into computer peripherals. These would include such software-bearing items as:

- Telephones (cordless, corded with electronics, cellular – essentially all phones sold today), which are used as computer input devices for phone menus, whether input is touch-toned or spoken. Similarly, voice mail systems are implemented with general purpose computers and use telephones as input/output devices.
- Hotel doorknobs, which use magnetic cards or other devices to send request for entry into a general purpose computer in the hotel office.
- Smoke detectors, fire alarms, and other such safety critical devices which in large buildings are often monitored by a general-purpose computer.
- Laptop computer batteries that have integrated monitoring chips to provide charge level information to the laptop computer. While part of a computer, they are hardly what is normally thought of as a "peripheral device."
- Home medical devices. This is a key example of a product whose software should (we think) always be subject to Article 2 but which might fit under UCITA. The product is Johnson & Johnson's SureStep Blood Glucose Monitoring System.[8] Initially this device had very limited functionality. Prick your finger (to draw blood). Wipe the blood on a test strip. Insert the test strip into the LifeScan reader and the device presents you with a reading of blood sugar level. More recent versions of the device come with diagnostics. More recent versions of the device store the last several readings. Now, you can download software, get a cable, and transfer data from the monitor to your personal computer. One version of the device, which sells for under $100, is available at many pharmacies and does not require a prescription, stores up to 150 tests and automatically creates 14-day and 30-day test averages. A newer model still, the Profile, can hold 250 tests, it records additional information about the user (such as insulin type and dosage), labels tests by events (such as exercise), thus creating fairly detailed database records that carry an activity code, a reading, a time stamp, a date stamp, and probably other information. This can all be downloaded to a computer. The value of this device lies primarily in its software. If that software is not upgradeable by download from a connected computer, we believe it would be easy to add this to the design. The manufacturer could easily create a separate license for the software (just toss another piece of paper in the book and add a click-wrap function to be executed on first use). Additionally, the manufacturer could offer optional software that can be uploaded to the device. Like the sewing machine, these software options could easily be made separately available.

A further point that must be addressed by the term "computer peripheral" is whether a peripheral's software is excluded from Article 2 only when it is actually being used as a peripheral, or whether it has ever been used as a peripheral, or whether by its manufacture it might someday be used as a peripheral.

### *2.3 Is this a peripheral?*

- **A printer** connected directly to a computer is a peripheral. Is a printer connected only to a network a peripheral? (Even if it is exactly the same model printer?) Is a copier that can be used as a printer a peripheral?

- **A digital camera** connected to a PC occasionally to download pictures might sometimes be a peripheral. But a digital camera in which "digital film" is removed and placed in a special reader is never connected to a computer – is it still a "peripheral."
- **A digital camcorder** has the capability to transfer video to a PC, but may well never do it depending on usage since it can equally well store video on tapes.
- **Noise-canceling earphones** have computer software to perform noise cancellation and are equally usable for listening to music played by a computer or a non-computer stereo system. If sold with a computer are they a peripheral? What about if they're sold with a stereo set? What if they're sold in a computer store but used with a stereo, or used with both?

### *2.4 Is this a computer?*

- **Digital voice recorder.** This is a solid-state replacement for a personal/dictating tape recorder. High-end models use voice recognition for completely button-less operation. This is a computer with memory that accepts user input, records data, compresses the data, then performs decompression and playback at a later time.
- **Calculators.** The line between calculators, PDAs, and other such devices is quite blurry and has been for more than 20 years, ever since the introduction of the TI-59 programmable calculator with magnetic tape storage in the late 1970s.

### *2.5 When does this become a computer?*

Some types of equipment have varying points upon a price-performance continuum in which some products are arguably computers and some are not. Where is the line drawn?

**Case 1: scanners.**

A low end scanner can be purchased for about $100 or less. This connects with a cable to a computer and is clearly a "peripheral".

Consider the HP 9100C digital sender, a high end scanner. This scanner can be used, out of the box, in the following ways:

- The scanner can be connected to computer network to send scanned documents as e-mail attachments to users worldwide. The system boots from a hard disk. A QWERTY keyboard and monochrome LCD display are used to enter e-mail addresses, display menu choices, and so on. In this mode it behaves like a computer with a scanner attached (in fact, it is really a "computer" + scanner sold as a single item).
- The scanner can be used solely as a high-capacity fax machine.
- The scanner can be used solely as a copy machine by connecting it via a dedicated network to a printer (just scanner and printer, no distinct "computer" involved). This forms a copier, with neither the scanner nor the printer being computer "peripherals". In fact, some copiers currently on the market are assembled out of precisely these pieces sold as a "copier" and can be optionally networked.
- The scanner can directly access a file system on a remote computer and perform a file transfer, just as is done with computer-to-computer file transfers, and even put documents directly on a web server.
- The scanner can do all of the above at once. It currently costs about $3000; does that make it a computer? Will it be just a peripheral when it only costs $100?

**Case 2: Cell Phones**

Simple cell phones are arguably neither computers nor peripherals. But most cell phones have LCD displays to display phone numbers. When does it become a peripheral/computer?

- When it displays outgoing calls with the numbers shown by the on-phone computer?
- When it is used to generate touch tones as input to a computer at the other end of a voice connection? (Serving as a computer input and output device makes it a "peripheral" by most definitions.)
- When it displays incoming pager messages?
- When it displays incoming e-mail messages sent via a computer exploiting an alphanumeric paging protocol?
- When it displays incoming e-mail messages sent via a specific e-mail function?
- When it can send simple text messages (called SMS for Short Message Service, deployed in Europe)?
- When it displays web pages?
- When it can generate simple web pages and send them out?
- When someone can do everything with a phone that can be done with a PDA (arguably a PDA is a computer)?
- When the phone is a PDA platform plus a cell phone system?

At some point cell phones pass from being cell phones to "peripherals" to "computers" under Article 2. "Computer"-class cell phones already exist. One can argue there is a trend for anything that can communicate with computers to evolve into a computer itself. One can argue that every embedded system that can be networked will eventually have a web server built in, and that most embedded systems will be networked. Is UCITA really intended to eventually cover everything?

When a car engine can send out a simple web page with status information to a mechanic, is it now a "real" computer instead of an "embedded" computer? Industrial equipment such as thermostats already do this today; the rest is simply a matter of time.

### *2.6 Use of "peripherals" for non-computer purposes*

If a hard disk is used in a non-computer role, does it remain a peripheral? Hard disks themselves have one or perhaps several CPUs, each executing software from fixed or reprogrammable memory. As an example, hard disks can be found in the following non-computer applications:

- Stand-alone scanning stations (which may or may not be "peripherals" depending on how they are used).
- Compressed audio, stand-alone jukeboxes (convert CD audio to MP3 format and play from a hard disk).
- Some newer digital cameras.

Some items can be used as either peripherals or as stand-alone devices, with no clear preference or intent for use by the manufacturer with respect to being just in one role or the other. Which of the following is a peripheral, and which is not? (And does it matter how it is being used or what its use history has been?)

- All-in-one printer/copier/fax machines. They may run stand-alone as copiers/faxes, or be connected to a computer as printers/scanners, or some combination thereof. There is no single preferred and obvious usage.
- High-end digital copiers, which can be connected to a computer network to accept print jobs or perform as an actual copier (the difference between this and an all-in-one printer/copier is that one costs a few hundred dollars and the other costs a few thousand dollars)
- Cash registers
- Digital photograph printers (either direct connect to camera or connect to computer)
- Recording thermometer/hygrometers with paper charts plus digital outputs, that can optionally host a web page.

### *2.7 Inherently dual-use components*

Most computer "peripherals" contain or consist of components that themselves have software. However, many such components are inherently dual use in that they can be used in ways that are not at all computer related. In fact, computer peripherals are very often created by coopting non-computer technology to benefit from the economy of scale of mass consumer products. Historical examples of this include: teletypes (electro-mechanical devices) used as computer terminals, electric typewriters used as computer terminals, personal copier engines used as the basis for laser printers, CD audio drives used as CD-ROM storage devices, and DVD drives used as DVD-ROM storage devices.

The fact that some components are inherently dual-use, and that the primary use by volume may not be as a computer peripheral, leads to a dilemma for the vendor producing the component. Should any software license be worded to fall under UCITA or not? For example, should the software controlling the head position and rotation of a CD transport mechanism be considered to be within UCITA because it might possibly be used for a computer? What if it is originally sold to be used in a CD audio application and it is later purchased by a company that instead uses it within a computer (or the other way around?)

Of course the case of dual-use components is in some ways similar to entire dual-use products. But, the situation is worse in that manufacturers of dual-use components may have no practical way to know how their components will ultimately be used.

# 3. Policy Problems

UCITA is generally more vendor-favorable than Article 2. Policy differences led to the break-up of the Article 2 project into separate Article 2 and Article 2B (UCITA) projects. If Article 2 excludes the coverage of embedded but non-integrated software from its scope, we expect that courts will look to UCITA for guidance, even in jurisdictions that have not adopted UCITA.

In this section, we will note only one policy problem, as an example of the incentives to a manufacturer of goods who wishes to do business under UCITA rather than Article 2. That example is the interference with the market for used goods

Proposed Section 2-103 (e) states:

> *Nothing in this article invalidates an otherwise effective provision in an agreement precluding a purchaser from selling or otherwise transferring a copy of a computer program or otherwise limiting in any manner the buyer's right to use the copy. However, a term in a consumer contract limiting the buyer's right to sell or otherwise transfer an integrated product is ineffective if the buyer transfers the integrated product, including*

*any incorporated computer program, without removing the program or retaining a copy of it.*

The manufacturer who can design its embedded software to be non-integrated within the meaning of Article 2, can restrict transferability of its products. The manufacturer who does business solely within Article 2 cannot.

Imagine buying a car that has non-integrated software to control the brakes, fuel injectors, suspension, steering, maintenance diagnostics, and other key functions. What will the resale value of the car be if the customer can be precluded from selling the embedded (if non-integrated under Article 2) software along with the car?

# 4. Likely Circumvention Mechanisms

A few of the possible mechanisms to circumvent exclusion of what would otherwise be embedded systems are outlined below. This is not meant to be an exhaustive list, but rather merely a foretaste of things to come should Article 2 exclude non-integrated software, throwing the handling of these types of embedded software to UCITA or to some other non-uniform body of law.

### *4.1 Pseudo-mandatory upgrades*

Software integral to an embedded system might not be excluded from Article 2. However, without sweeping changes it is clear that pure software upgrades sold to improve functionality of a system would be excluded. This could be exploited by selling, for a low fee if necessary, after-market software upgrades. Examples include:

- Providing an automotive engine controller with mediocre performance as standard equipment, and requiring the specific purchase and license of an “enhanced performance software upgrade”
- Shipping DVD players with an initial version of software with significant, known defects, and providing upgraded software to users who complain about problems or alternately offering a software upgrade as a separately billed item at the time of sale (perhaps with a sales promotion discounting retail price by that amount, etc.). (Note: it is already common to “repair” DVD players having trouble playing a particular movie title by installing a software upgrade.)
- Using the permitted provision to limit the length of life of a license to a very short period of time, then requiring an upgrade to extend the license (perhaps with an upgrade feature set including unlimited length license).

### *4.2 Sell formerly embedded functionality as an access contract*

Rather than embedding functionality in a product, a manufacture can provide it remotely. If this is done within UCITA, the manufacturer also evades its product's “mass market” status (because this is an access contract). The risk of remote supply of services is reduced effective reliability of embedded products. As the Internet becomes available, this business model is gaining popularity and has merit in situations such as digital television recorders that update program listings each night. However, this approach could be invoked artificially merely to take software out of Article 2 (and allow the vendor to opt the full product into UCITA) if the system has networking or modem ability as a standard capability. Note that it is possible to have an access contract model where a single payment of low value purchases an access subscription for the life of a piece of equipment (this is already done in some digital video record/playback systems). Possible

scenarios include those below. It is becoming increasingly common for PC-based software to automatically perform on-line updates, so this sort of business model seems quite likely for anything with a network and reprogrammable memory in the future.

- Digital television recorders that only know how to capture programs based on listing information (via access contract and a modem) and have no provision to enter specific date and time for recording.
- Devices that use Java applets (dynamically downloaded software) instead of built-in software, which can be a preferred method of keeping software updated in Java development.
- Adding engine control functionality to an automobile that is responsive to the road conditions and expected weather conditions based on position to improve performance, but that require an information download. This sort of functionality is in fact built into high end vehicles using in-vehicle sensors, but could be provided via cell-phone links using existing integrated cell phones and navigational position information instead and save the cost of the vehicle sensors to boot.

### *4.3 Sell software as a household site license*

Site licenses receive even less protection than other software. Software could be sold as a "site license" to individual residences only, thus leaping out of the status of "mass market". While such an arrangement might be considered generous to a software purchaser who can now run the software on more than one computer if installed at home, for many consumers this will be meaningless (since they have only one computer) but will compromise their rights under UCITA. In some embedded systems, software will be distributed across household internet appliances, requiring a site license for a house in order to use mobile applets to coordinate appliances and so on. This will force such systems into a special site license category rather than making them mass market products, although it would seem unreasonable to have a distinction made as to whether a piece of software is implemented as centralized or distributed software be the primary source of determining whether it is a mass market product or not.

### *4.4 Artificially expose computer access*

It is often in the best interest of all involved to hide general purpose computer capability from a user if that computer is being used as a dedicated embedded system. Since UCITA preemptively includes all general-purpose computer software, and Article 2 will exclude such software, throwing judges to UCITA even when UCITA has not been adopted in the jurisdictional state, there is significant incentive for application developers to develop upon a general-purpose operating system (which they might well have been doing anyway), and artificially expose the functionality of the general purpose system to the user even if it is never seriously intended to be used. For example, a DVD player might run upon an operating system and might display a "click here" box upon power up to change operating modes into the raw operating system. If it were theoretically possible to load some other software into the DVD player to run it (which would be easy to arrange), then that would convert the DVD player into a "computer", bringing it within the scope of UCITA. Given an increasing trend to use off-the-shelf operating systems in embedded products, this might apply to a surprising number of things that would otherwise be non-computer goods.

# 5. Conclusion

The proposed language of Article 2 will be ineffective at distinguishing between embedded and non-embedded software. By using flash memory and providing separately sold software options, a manufacturer can easily take the software components of traditional goods out of Article 2.

Even if the language of Article 2 and the official comments were to be changed to more effectively exclude embedded computer systems, embedded and non-embedded computing technology are expected to converge in the next few years, making the problem more difficult over time.

---

[1] Bill Henning and Henry Gabriel, *Summary of Scope Proposal*, memo to Members and Observers, Article 2/2A Drafting Committee, dated November 13, 2000. All quotations attributed to the revised Article 2 draft are from this memo.

[2] NOTE: This draft is being circulated at the St. Louis meeting of the Article 2 Drafting Committee on November 18, 2000. This circulating copy has most recently been revised by Cem Kaner and revisions have not yet been approved by Phil Koopman. Any errors or omissions are the fault of Kaner and not Koopman.

[3] Philip Koopman is as an Assistant Professor at the Carnegie Mellon University Electrical and Computer Engineering Department, with an additional appointment to the Institute for Software Research, International (ISRI) in the School of Computer Science. He is also the Embedded and Reliable Information System (ERIS) thrust leader at the Carnegie Mellon Institute for Complex Engineered Systems, and the Dependable Embedded System thrust leader for the CMU/General Motors joint research laboratory.

Koopman has previously held positions with various users and creators of embedded computer systems, including: U.S. Navy Submarine officer (gold dolphins; cold war veteran), U.S. Navy Engineering Duty officer (weapon system integration and lifecycle support), Harris Semiconductor Senior Scientist (embedded computer CPU/software/firmware/compiler research and development), and United Technologies Principal Engineer (embedded system R&D for elevators, jet engines, radars, sonars, climate controls, automotive subsystems, and helicopter applications). Among the commercial software products he has developed are cryptographic security algorithms for automotive systems, manufacturing software, and a compiler.

Koopman's research interests include quantifying software robustness, developing gracefully degradating system architectures, and analyzing real time embedded network protocols. Among his research results are a tool (the Ballista testing system) that can automatically find one-line programs that crash commercial operating systems and other software packages. His primary teaching emphasis is on practical methodologies to create robust embedded systems and software.

In addition to his Carnegie Mellon responsibilities, Koopman is an Associate Editor of Design Automation for Embedded Systems: an international journal, and is the chair of the IFIP working group 10.4 SIG on computer system dependability benchmarking. He has written four books, the usual assortment of papers, and holds twenty U.S. patents in embedded system technology and applications. He received a Ph.D. in Computer Engineering from Carnegie Mellon University in 1989 and both a M.Eng. and B.S. in Computer and Systems Engineering from Rensselaer Polytechnic Institute in 1982.

[4] Cem Kaner is Professor of Computer Sciences at the Florida Institute of Technology. He is senior author of *Testing Computer Software* (2nd Ed., 1993), which is the best-selling book in the software testing field's history. He is also senior author of *Bad Software: What To Do When Software Fails* (1999) and the author of over 100 other publications. Kaner holds a J.D. (Golden Gate University School of Law, 1994), a Ph.D. in Experimental Psychology (McMaster University, 1984) and a B.A. in Arts & Sciences (primarily Math and Philosophy, Brock University, 1974). Kaner is also Certified in Quality Engineering by the American Society for Quality. He is also a member of the American Law Institute.

[5] References to the Uniform Computer Information Transactions Act are to the draft of September 29, 2000, downloaded from www.law.upenn.edu/bll/ulc/ulc.htm on November 18, 2000.

[6] IEEE Std 610.10-1994 is the Institute for Electrical & Electronics Engineers *Standard Glossary of Computer Hardware Terminology*. IEEE Std 610.12-1990 is the IEEE *Standard Glossary of Software Engineering Terminology*.

[7] Turbo City Performance Headquarters, *Hey There Corvette Crossfire Owners!* <www.turbocity.com/CorvetteCrossfileECMUpgrade.htm>. Checked 9/11/00.

[8] For current information on this product line, go to www.lifescan.com. Checked 9/11/00.