Student-centered Educational Tools
for the
Digital Systems Curriculum

Steven F. Barrett, Daniel J. Pack, George W. P. York,
Pamela J. Neal, Ruth D. Fogg, Edward Doskocz, Scott A. Stefanov,
Parris C. Neal, Cameron H.G. Wright, Alan R. Klayton

Department of Electrical Engineering, 2354 Fairchild Drive, Suite 2F6
United States Air Force Academy, Colorado 80840-6236
Voice: (719) 333-3190, FAX: (719) 333-3756,

ABSTRACT

The Digital Systems Division's mission within the Electrical Engineering Department at the United States Air Force Academy is to educate cadets on the fundamentals of digital systems. The division provides a digital systems curriculum to computer science and electrical engineering majors. Additionally, we teach the fundamentals of microcomputer programming to all electrical engineering majors. Over the last four years we have implemented a variety of in-house hardware and software teaching tools to enhance our educational mission while emphasizing an exciting, hands-on approach to computer education. In this paper we will detail these innovations and describe how they fit together for a cohesive educational experience.

BACKGROUND

The digital systems curriculum at the United States Air Force Academy includes five different courses: Introductory Digital Systems (EE281), Microcomputer Programming (EE382), Microcomputer System Design I (EE383), Microcomputer System Design II (EE484), and Computer Architecture (EE485). This sequence of courses serves several different audiences. All electrical engineering majors are required to complete the first two courses in the sequence. Majors may also elect to complete a digital systems option within the electrical engineering major by completing two more courses in the sequence. Computer Science majors are also required to complete Introductory Digital Systems and may elect to complete a digital systems option by taking at least the first three courses in the sequence. A brief summary of course content follows:

- **EE281 Introductory Digital Systems.** This sophomore level course provides an introduction to the fundamental principles of logic design including: Boolean algebra, combinational and sequential logic analysis and design, and an introduction to digital computer architecture.

- **EE382 Microcomputer Programming.** This junior level course focuses on assembly language programming while providing a broad-base understanding of microcontroller systems. The microcontroller principles presented provide a foundation that can be used in other project-oriented courses. Course topics include microcontroller hardware, assembly language programming, input/output interface design, and applications.

- **EE383 Microcomputer System Design I.** This junior level design course provides instruction on the design of digital systems using microcontrollers. Topics include structured system design, instruction sets, support software, system timing, input/output, peripherals, and interfacing techniques.

- **EE484 Microcomputer System Design II.** This senior level course is the culmination of the digital systems design sequence using microcontrollers. Students investigate advanced peripheral interfacing techniques, memory systems, bus features, coprocessors, serial communications, and digital to analog conversion techniques.

- **EE485 Computer Architecture.** This final course in the digital systems area of study quantitatively examines trade-offs in the design of high-performance computer systems. Topics include price/performance, instruction sets, hardwired control versus microprogramming, memory hierarchy, cache memory, virtual memory, pipelining, reduced instruction set computers, input/output, and parallel processing.

In 1993 the electrical engineering department head charged the division to investigate and develop innovative, hands-on teaching tools for the digital systems curriculum. The purpose of these tools were to address and correct self-identified limitations within our curriculum. Some of the identified limitations included:

- In the EE281 Introductory Digital Systems course an EDUCOMP (EDUcational COMPuter) trainer was used to teach the interrelationship between computer hardware and software. This trainer was designed and built in-house in the mid 1970s. The trainer had served long and well; however, its memory was restricted to 16 address locations which limited instructional opportunities and reduced cadet motivation.

- In the EE281 Introductory Digital Systems course a circuit simulator such as Micro Sim's Evaluation PSPICE was used to simulate student laboratory project designs prior to implementation. This is an important step in the design, simulate, build, and test process. Precious classroom time was being used to teach the fundamentals of the PSPICE simulation environment. Furthermore, considerable instructor office time was spent answering many basic questions about the PSPICE simulation environment.

- In the EE383 and EE484 Microcomputer System Design course sequence a single board computer (SBC) concept was used to teach microcomputer design and programming techniques. Both courses consisted of lectures on specific microcomputer hardware or software concepts followed by several laboratory classes in which the circuit under study was built in the laboratory using breadboard and wirewrap construction techniques. Using this technique an entire SBC was designed and fabricated during the course sequence. This concept worked well as long as students did not encounter any significant problems in their design. If design or fabrication problems occurred, students often fell behind on subsequent laboratory exercises.

- In the EE382, EE383, and EE484 microcomputer design sequence, students had limited access to laboratory equipment and software during prime evening study hours. This limitation was due primarily to the digital design laboratory having only 24 stations. This did not pose a problem during course instruction since class size was limited to 24 students or less; however, during peak study times, particularly when big projects were due, laboratory space was at a premium.

- Overall, we wanted to spark enthusiasm in the curriculum and make learning computer design fundamentals a rewarding experience for our students.

METHOD

To address these concerns, the faculty and staff of the division developed a number of student-centered educational tools. These tools evolved over a four year period. Each tool was carefully considered, developed, implemented, and tested prior to incorporation into a course. Each of these tools is detailed below.

VISICOMP. The VISIble COMPuter or "VISICOMP" was designed, as was its predecessor EDUCOMP, to visibly show all major computer systems operation with software driven stimulus. VISICOMP may be run in the automatic mode at a user-selected speed or in the manual mode. In the manual mode the user provides all system control signals via front panel switches. VISICOMP was designed in-house in 1993. A wire wrap prototype of this new computer concept trainer was completed by one of our own electrical engineering graduates in Fall 1994. Full scale production of the VISICOMP units was completed in Fall 1995. VISICOMP was introduced to the classroom in Spring 1996.

VISICOMP is comprised of five subsystems each on a separate printed circuit board: the controller, the arithmetic logic unit , the input/output subsystem, the memory subsystem, and a motherboard. Each board is populated with medium scale integration components and numerous light emitting diodes (LEDs) and seven segment displays. The VISICOMP features a fully visible data bus, address bus, and control bus as well as visible indicators of key register contents, control signals, and the controller states. It also contains 256 nibbles (4 bit data words) of memory and 16 operation codes. A small monitor program resident on system ROM facilitates user program loading and debugging. Reference Figure 1 and 2.

The last 12 lessons of the EE281 Introductory Digital Systems course is used for VISICOMP instruction. Here the students review the operation and application of digital hardware learned previously in the course. Furthermore, through a series of hands-on laboratory exercises students learn assembly language and machine language programming concepts, computer architecture design and operation, and most importantly the opportunity to pull all course concepts together as they relate to the design of a digital system.
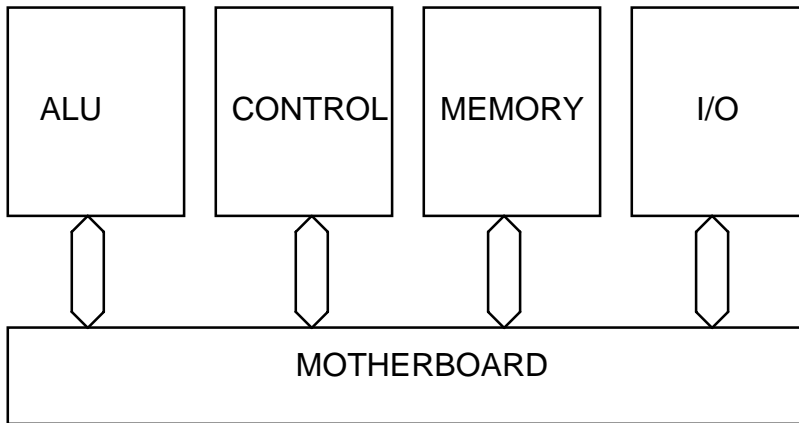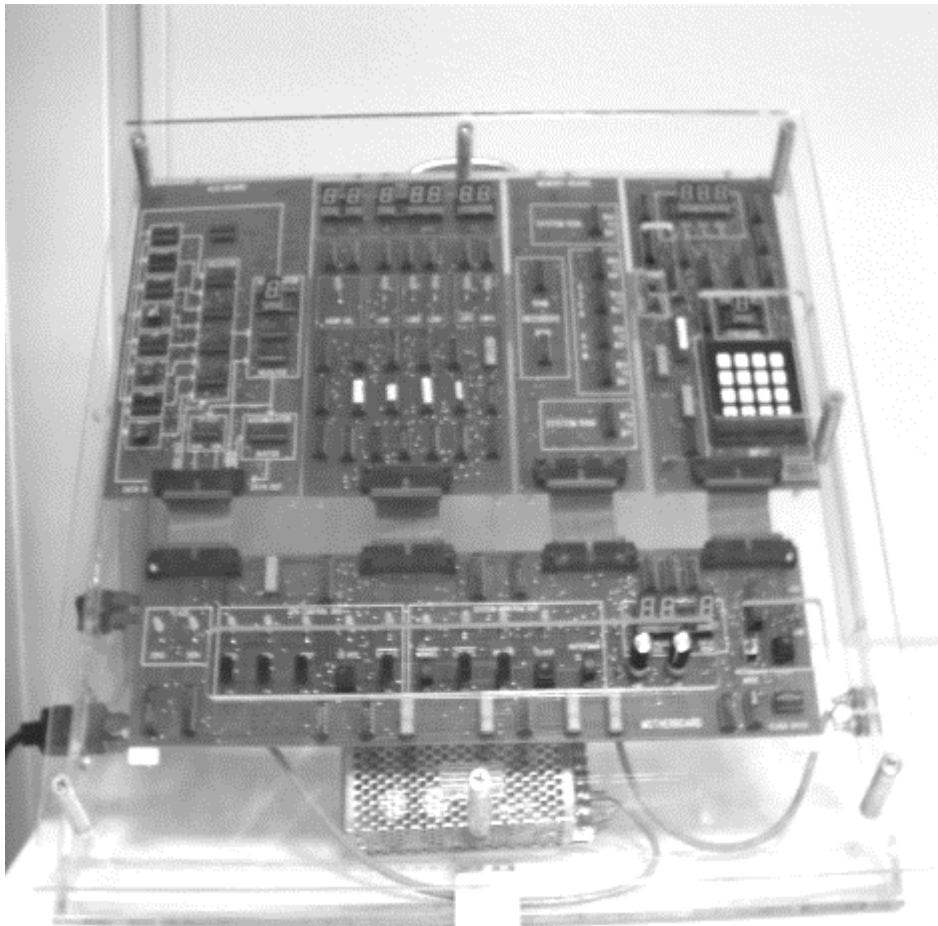
Figure 1. VISICOMP Computer Architecture

Figure 2. VISICOMP

VISICOMP Software Simulator. To allow ready student access to VISICOMP programming aids, a Java VISICOMP emulator was developed. This simulator was designed to look and operate just like its hardware namesake. It is provided to students over the university-wide computer network. It provides the opportunity to practice and develop assembly language programming skills without needing physical access to VISICOMP. Although designed as a programming tool to enrich the VISICOMP educational experience, this simulator may be used as an instructional aid in a digital design course if funds are not available to build enough hardware-based computer trainers. It may also be used to test software ideas before actually loading a program on the hardware. The VISICOMP emulator screen is provided in Figure 3. The VISICOMP instruction set is provided in Table 1.
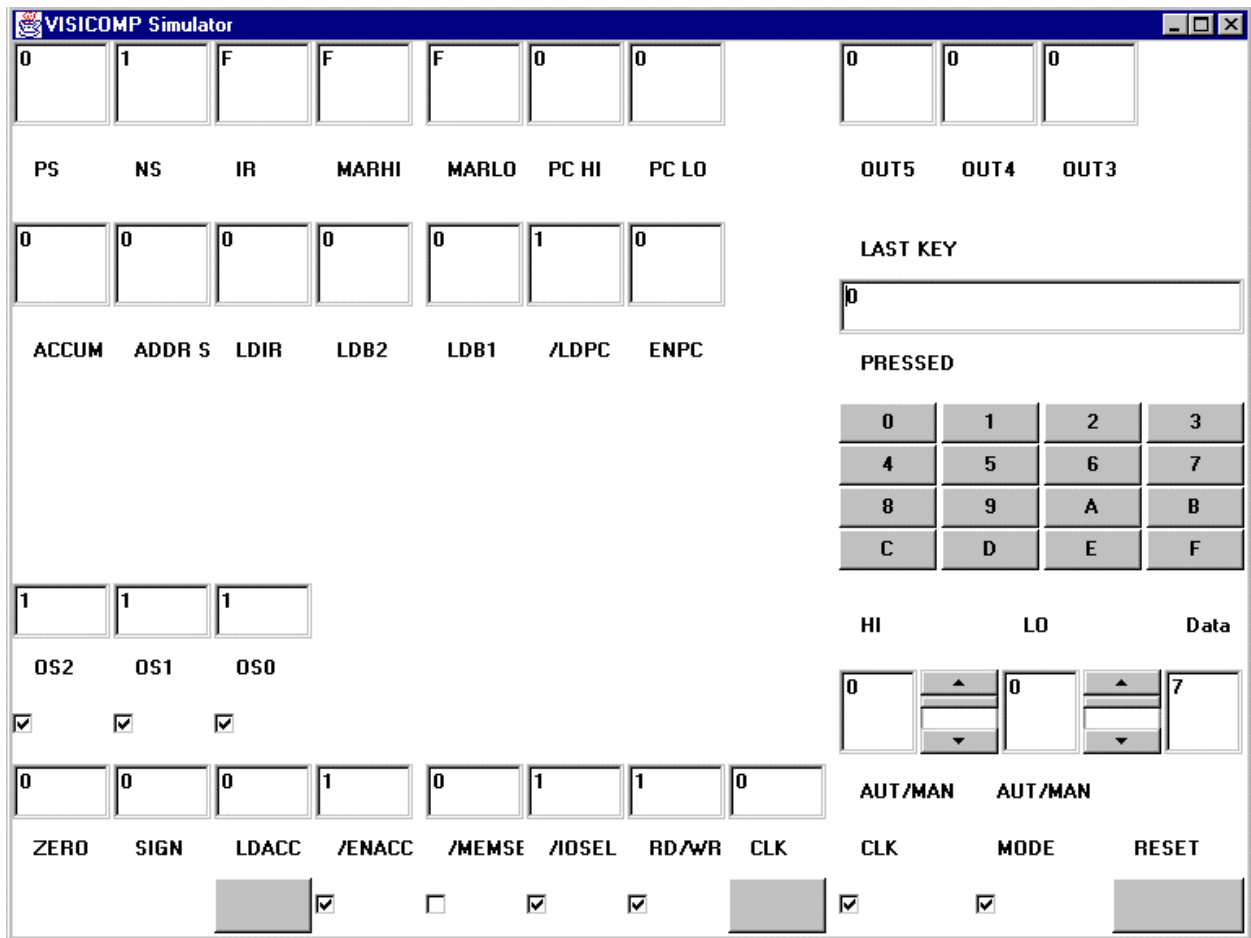

Figure 3. VISICOMP Simulator

PSPICE Tutor. A 20 minute PSPICE Tutor was developed using Microsoft's Camcorder software. Camcorder is a screen capture package that digitally records all activities on a PC screen until it is turned off. Camcorder also has the capability to record an audio track along with the screen capture. The output file is a self-executing compressed file that operates in the Microsoft Windows 95 environment without the need to install any host system!

The tutorial was developed by activating Camcorder and then building a PSPICE digital circuit simulation using normal Micro Sim Evaluation PSPICE software. The instructor explained all of his actions while completing the simulation steps. Thus on playback, the student not only sees all of the instructor's on-screen actions, but also hears the associated explanations. The tutorial was placed on a student accessible drive on the university-wide network. Instructors provided loading instructions to the students. Students were told instructors would be available to answer questions only after students had viewed the tutorial. The results were quite dramatic. No class time was required to teach PSPICE operation and extra instruction provided during instructor office hours was reduced to a minimum. Any PC-based process can be demonstrated with this technique.

INST.	BINARY	HEX	OPERATION
ONE WORD INSTRUCTIONS			
NOP	0000	0	NO OPERATION
NEG	0001	1	NEGATE THE ACCUMULATOR (2'S COMPLEMENT)
NOT	0010	2	COMPLEMENT THE ACCUMULATOR (1'S COMPLEMENT)
RAR	0011B	3	ROTATE ACCUMULATOR RIGHT
TWO WORD INST. - SECOND WORD IS I/O PORT OR IMMEDIATE VALUE			
OUT	0100	4	OUTPUT ACCUMULATOR TO ADDRESSED PORT
IN	0101	5	INPUT ADDRESSED PORT'S VALUE TO ACCUMULATOR
ADDI	0110	6	ADD IMMEDIATE VALUE TO ACCUMULATOR
LDAI	0111	7	LOAD IMMEDIATE VALUE TO ACCUMULATOR
THREE WORD INST - 2ND AND 3RD WORDS ARE LO AND HI-ORDER ADDRESS			
AND	1000	8	AND ACCUMULATOR WITH VALUE AT ADDRESS
JMP	1001	9	UNCONDITIONAL JUMP TO ADDRESS
JZ	1010	A	JUMP TO ADDRESS IF ACCUMULATOR = 0
JN	1011	B	JUMP TO ADDRESS IF MOST SIGNIFICANT BIT = 1 (NEG)
OR	1100	C	OR ACCUMULATOR WITH VALUE AT ADDRESS
STA	1101	D	STORE THE ACCUMULATOR TO ADDRESS
ADDD	1110	E	ADD VALUE AT ADDRESS TO ACCUMULATOR
LDAD	1111	F	LOAD VALUE AT ADDRESS TO ACCUMULATOR

Table 1. VISICOMP Instruction Set

PORTOLAB. To provide ready student access to microcomputer laboratory equipment, a portable, self-contained microcomputer laboratory was designed and fabricated in-house. We've dubbed this student trainer PORTOLAB. The PORTOLAB contains a Motorola 68HC11 evaluation board (EVB). The EVB is a single board microcontroller unit featuring an eight bit data bus and a 16 bit address bus. In addition to the EVB, the PORTOLAB contains a system power supply and a small protoboard to support interfacing of external hardware to the 68HC11. A storage area inside the PORTOLAB lid contains a logic probe, wiring kit, power cord, and interface cable to connect the EVB to a host personal computer containing the software development system. The entire PORTOLAB was built into a 12" x 10" x 6" lunch box sized black hard plastic case (Pelican Products Incorporated, Torrance, California). Reference Figure 4. The PORTOLABs are maintained in-house by a staff technician.

Figure 4. ROBOLAB with PORTOLAB

The PORTOLAB was originally developed for use in our EE382 Microcomputer Programming course. A PORTOLAB is issued to each student at the beginning of the semester. The students are required to bring the PORTOLAB to scheduled course laboratories. The PORTOLAB allows the student to complete their pre-laboratory exercise in their dormitory rooms, bring the PORTOLAB to scheduled laboratory periods, and then complete, if necessary, any remaining laboratory requirements back in their dormitory room. (All students have a PC in their rooms equipped with a 68HC11 development system to develop, assemble, and download programs to the 68HC11.)

The PORTOLAB concept has proven so successful it has also been adopted for use in the EE383 and EE484 Microcomputer System Design course sequence. With the replacement of the single board computer concept in these courses with the PORTOLAB, precious laboratory time is now used to teach more microcomputer concepts as opposed to spending time wirewrapping SBC components. Students still have the opportunity to learn wirewrapping techniques in a memory expansion laboratory exercise.

Provided below is a partial list of laboratory exercises completed by students along with key microcontroller concepts applied in the exercise using the PORTOLAB in the EE382, EE383, and EE484 microcomputer design sequence:

- Electronic Dice - external hardware interface techniques and basic interrupt processing,

- Intersection Stoplight - external hardware interface techniques, internal timers, advanced interrupt processing,

- Frequency Divider - advanced input and output features of the microcontroller,

- Digital Voltmeter - analog-to-digital conversion (ADC) features and liquid crystal display (LCD) interface,

- Frequency Counter - multi-character seven segment display interface,

- Pong Game - LCD text and graphic display interface and serial mouse interface,
- Calculator - advanced interrupt techniques and keypad interfacing,

- Digital Filtering - Digital-to-analog and ADC conversion techniques.

ROBOLAB. ROBOLAB is an in-house developed, motorized robotics platform used with PORTOLAB in the EE382 Microcomputer Programming course to teach assembly language programming techniques and microcontroller hardware concepts. The director of the course introduced the ROBOLAB concept to achieve course objectives of teaching assembly language concepts, microcontroller hardware components, and interfacing techniques in a motivational atmosphere. The director also wanted to focus the course on an overall, integrated course project rather than a series of disjoint laboratory exercises.

The ROBOLAB consists of an eight inch circular, aluminum platform riding on two 2.5" diameter wheels each of which is driven by direct current motor. Reference Figure 4. The platform is also equipped with three infrared transmitter/receiver sensor pairs. These sensors are used to detect obstructions. The ROBOLAB was introduced in the microcomputer programming course in the Fall 1997 semester. Students were issued the platforms early in the semester. During the course of the semester the students interfaced a LCD panel display to the ROBOLAB as a status indicator, developed all required hardware and software to steer the ROBOLAB through a maze, and implemented the motor control hardware and software for controlling the platform's wheels.

The culmination of the course was a robotics navigation maze competition. Each student's robot competed against the others and also the course director's robot. The winning robot was determined based on the shortest amount of time to navigate through the maze and successfully exit through a pre-selected maze door. A student's score was weighted depending on the exit door chosen. Exit doors chosen further from the maze's entrance increased the difficulty of the exercise and thus enhanced the student's score. Points were deducted for wall collisions.

RESULTS

Developing and implementing these hands-on instruction tools required considerable time and expense. However, we feel that all of these innovations were well worth the effort. Here is a summary of benefits derived from implementing these tools:

- With VISICOMP students were able to clearly tie the concepts of hardware and software together. They were also able to "see" how all course concepts fit together in the design and operation of a computer. Most importantly when the student acted as the system controller when using the VISICOMP in the manual mode, they could clearly see how the control signals effected system operation.

- The Java VISICOMP Simulator allowed students to practice their assembly language programming skills and follow the control signals through a computer system without access to VISICOMP hardware. Furthermore, it allowed students to practice programming skills in their dormitory room.

- PORTOLAB provided each student ready access to laboratory equipment. Furthermore, it allowed students to effectively use their time when preparing and completing laboratory exercises. PORTOLAB also provided a self-contained, easy to use platform for completing many motivational laboratory exercises specifically designed to teach microcontroller concepts.

- The PSPICE Tutor provided students a readily available introduction on PSPICE digital simulation techniques. Furthermore, it freed up valuable classroom time and instructor office time.

- ROBOLAB sparked student enthusiasm and competition while learning assembly language programming, microcontroller hardware, and input/output design skills. It also provided a mechanism for students to incorporate multiple functions of a microcontroller into a single project.

- Development of these tools was truly a team effort (note number of co-authors). As we worked to design, implement, and fabricate these educational tools; department cohesiveness and morale were enhanced. Furthermore, development of these tools served as a good professional development exercise for the department's faculty and staff.

CONCLUSIONS

Although we have a good complement of self-developed tools to aid in the instruction of digital systems concepts, we are constantly looking for new ideas. In the next year we are working to incorporate a "C" cross compiler into our Microcomputer System Design course sequence, developing a computed-based data acquisition and instrumentation course, and increasing the role of advanced programmable logic (e.g. Field Programmable Gate Arrays (FPGAs) in the curriculum.

REFERENCES

D. Pack, G. York, P. Neal, and S. Stefanov, "Constructing a Wall-Follower Robot for a Senior Design Project", 1996 ASEE Annual Conference and Exposition, June 1996, Washington D.C.

G. York and R. Fogg, "VISICOMP: The Visible Computer", 1996 ASEE Annual Conference and Exposition, June 1996, Washington D.C.

G. York and P. Neal, "MC68HC11 Portable Lab Unit - A Flexible Tool for Teaching Microprocessor Concepts", 1996 ASEE Annual Conference and Exposition, June 1996, Washington D.C.

ACKNOWLEDGMENTS

AUTHORS

Lt Col STEVEN F. BARRETT, PhD is a Professor in the Department of Electrical Engineering at the U.S. Air Force Academy. His research interests include automated laser surgery, image processing, and instrumentation. He is a member of IEEE (Senior Member), SPIE, and Tau Beta Pi. Email: barrettsf.dfee@usafa.af.mil

Dr. DANIEL J. PACK, PhD is an Assistant Professor in the Department of Electrical Engineering at the U.S. Air Force Academy. His research interests include robotics, automatic target recognition, and computer vision. He is a member of IEEE, ASEE, Eta Kappa Nu, and Tau Beta Pi. Email: packdj.dfee@usafa.af.mil

Captain GEORGE W.P. YORK is a PhD candidate at the University of Washington on USAF Academy sponsorship. He will return to the USAF Academy as an Assistant Professor upon completion of his PhD studies.

Captain PAMELA J. NEAL is a PhD candidate at the University of Washington on USAF Academy sponsorship. She will return to the USAF Academy as an Assistant Professor upon completion of her PhD studies. Email:

Major RUTH D. FOGG, PhD recently retired from active duty. Prior to her retirement she served as an Assistant Professor in the Department of Electrical Engineering at the U.S. Air Force Academy. Her research interests include optical computing and optical computer layout tools.

Major EDWARD DOSKOCZ is a PhD candidate at the University of Alabama on USAF Academy sponsorship. He will return to the USAF Academy as an Assistant Professor upon completion of his PhD studies.

Major SCOTT STEFANOV, MSEE, is an Assistant Professor in the Department of Electrical Engineering at the U.S. Air Force Academy. He is also a C-130 pilot. Email: stefanovsa.dfee@usafa.af.mil

Colonel PARRIS C. NEAL, PhD, is Senior Military Professor in the Department of Electrical Engineering at the U.S. Air Force Academy. His research interests include instrumentation, computer architecture, acoustics, and audio electronics. He is a member of ASEE, IEEE, Tau Beta Pi, and Phi Kappa Phi. Email: nealpc.dfee@usafa.af.mil

Major CAMERON H. G. WRIGHT, PhD, PE, is an Assistant Professor in the Department of Electrical Engineering at the U.S. Air Force Academy. His research interests include signal and image processing, biomedical instrumentation, communications systems, and laser/electro-optics applications. He is a member of ASEE, IEEE (Senior Member), SPIE, Tau Beta Pi, NSPE, and Eta Kappa Nu. Email: cwright@tesla.usafa.af.mil

Colonel ALAN R. KLAYTON, PhD is Professor and Head of the Department of Electrical Engineering at the U.S. Air Force Academy. He is a member of ASEE, IEEE (Senior Member), Tau Beta Pi, and Eta Kappa Nu. Email: klaytonar.dfee@usafa.af.mil