Digital Communication Course With Computer Networking in EET

Asad Yousuf, Savannah State University
Venkatash Muniappa, Savannah Technical College
William Lehman, Advanced Systems & Materials

Abstract

Traditionally, digital communication at the Electronics Engineering Technology (EET)
Level, has focused on using amplitude modulation, frequency modulation, sideband communications, wave guide, radar and so on. However, with the advancement in technology the skills required to evaluate, integrate, and administer networking equipment is considered important for an EET student.

The popularity and acceptance of computer networking have created a need to provide students in Electronics Engineering Technology with a training course based on the aforementioned technology. Computer networking training can be introduced as an integrated approach to traditional communication topics and computer networking. In an effort to introduce students to computer network-based communication, the Electronics Engineering program at Savannah State University and Savannah Technical College has developed a course in which emphasis is directed towards a balanced curriculum that includes topics both from traditional communication area and computer networking concepts.

This paper includes the course outline and information for an innovative computer-networking laboratory. The laboratory facilities accommodate study of several computer networking hardware and software technologies.

Introduction

The large proliferation of communication and networking standards provides a daunting amount of material to present to the EET student in addition to presenting the traditional basics of communication theory. As with most areas of electronics even the so-called "traditional" basics of communication theory need greater emphasis due to innovation in smart antennas, radar technology, Radio Frequency Identification (RFID), wireless networks and other advances [3]. Lectures will at least touch upon some of the new concepts if for no other reason than to motivate the EET Student to learn more about these subjects.

EET Students may be involved in industry dealing with maintaining substantial networks of computers, each with significant amounts of complex hardware and software. The EET Student may also be faced with modifying or working at a detailed level on a given node of the network.

Due to the prevalence of the Internet, and the relatively low cost of setting up a complex TCP/IP network for a lab, TCP/IP is a good method of exposing the EET Student to a complex protocol.

For reasons of lab time, cost, and so on all of the standards that even exist today cannot be taught, much less standards under development. The network and specialized nodes can introduce the EET Student to a variety of networking concepts. Lectures and Labs provide comparisons of standards through a framework of tradeoffs associated with different networking standards and implementations such as maximum allowed distance between nodes, data bandwidth of node, bandwidth used by overhead of protocol, bandwidth average or guaranteed, etc.

Many of the new standards are based upon serial communication protocols transmitted on lines using differential signaling. To increase bandwidth, groups of serial lines are used in conjunction with one another to pass data from one IC or device to another IC or device. Lectures on single and differential serial communication standards will at least touch upon RapidIO, Fibre Channel, Infiniband, Advanced Telecom Computing Architecture (ATCA), Starfabric, etc [3,4,5]. Lectures will also cover parallel back plane standards such as ISA, VME and PCI, PCI Express.

Digital communication protocols, based upon concepts such as encapsulation of data, tunneling, CRC, and encryption are essential concepts needed to understand existing and future data standards. Other data communication concepts such as switching and data fault redundancy to reach a given node are being incorporated in the Advanced Switch standard (www.asi-sig.org) projected for 2005 as well as many existing Standards [3].

The Computer networking lab provides the students with the capability to deal with a real network of PC's running Windows connected on a TCP/IP network. The use of Embedded Networking Nodes provides simple devices with which the student has a chance to grasp the "nuts and bolts" of the TCP/IP protocol hardware and software. The Embedded Networking Lab exercises notes show state machine diagrams that capture the basic functionality of the device as used in the lab exercise. Lab exercises are meant to reinforce the connection between the state machine, protocol and design courses in Digital Design using VHDL, and Micro-Controller Design.

Using two Embedded Nodes that for a specialized purpose the class can be exposed by rotation and keeping the Embedded exercises short. A node with a micro-controller connected to a TCP/IP network, opens the door to numerous configurations and possible exercises. With a micro-controller network equipped with a TCP/IP interface the micro-controllers could pass data for example via RS-232, RS-422, RS-423, RS485, RS-644, I2C, USB, Serial Peripheral Interface (SPI), Controller Area Network (CAN), Local Interconnect Network (LIN) and a variety of digital standards. Some could potentially be configured to contain interfaces such as frame grabbers so can expose the EET Student to digital standards such as VoiceIP, IEEE 1394, MPEG, PAL, SECAN, NTSC, and SMPTE [1,2,3,4].

Some exercises will be included that involve badly behaving embedded Internet appliances to demonstrate the importance of reliability in embedded Internet appliances to network performance [9,10,11].

To avoid potential conflicts with several groups of students working on different sections of the network at the same time requires the introduction of bridge, routing and gateways to be introduced in the lectures before starting lab assignments.

Course Content

Electronics Engineering Technology students are introduced to the ELET 3701K Digital Communication course at the junior level. The course is a three-credit hour course consisting of two-hour lecture and a two-hour lab per week. The course covers the fundamentals concepts of Digital Communication and includes computer networking theoretical concepts and laboratory sessions. Students work in pairs on structured exercises, maintain detailed laboratory logbooks, and write formal reports. Students are exposed to the basic communication concepts and functions of networking during the entire course period. In addition to using conventional communication lab equipment, students are also exposed to the practical aspects of networking such as splicing cables, installing cable connectors, installing network interface cards, and configuring a network in different topologies. The laboratory also contains two embedded networking nodes to demonstrate embedded TCP/IP concepts.

The course is segmented in the following three areas:

- Introduction to Telecommunication
- Data Communications
- Computer Networks and the Internet
- Embedded TCP/IP Devices

<u>Introduction to Telecommunication</u>: includes topics such as:
Noise
Modulation AM, FM, PCM
Coding
Data Link Protocols
Telephony

<u>Data Communications</u>: includes topics such as:
Modulation
Line codes
OSI Model
Data compression
WANs protocols, ISDN, X.25, Frame Relay, ATM, SONET

<u>Computer Networks and the Internet</u>: includes topics such as:
LAN architecture and protocols
WAN architecture and protocols
Internetworking devices: routers, hubs, and switches
Wireless technologies
World Wide Web:HTML, web pages, servers, browsers

<u>Embedded TCP/IP Devices</u>: includes topics such as:
802.3 Frame
IP Encapsulation within 802.3 Frame
Bandwidth of TCP/IP
Overhead Associated with TCP/IP
Comparisons of TCP/IP to other Digital Communication Protocols

It is a daunting task to find a textbook that will cover all three areas. So to cover the entire aforementioned concepts instructor integrated notes are distributed to the students for Introduction to Telecommunication and Data Communication. However, a textbook in Computer Networking is used in the classroom setting.

The following paragraphs will describe a typical laboratory setup for the Embedded Node and Computer Networking Laboratory, along with a step-by-step guide to Client/Server Networking and Embedded Node exercises.

Computer Networking Laboratory Setup

The laboratory assignment first introduces a simple examination of a network and its physical and logical topology. Students are explained about the following practical aspects of networking:

- Installing Cable Connector
- Installing Network Interface Boards
- Configuring drives
- Setting up Network in different topologies
- Configuring Network Hardware

After the completion of the hardware setup of the network, students are required to install a network operating system (in particular windows 2000). They perform the following tasks:

- Peer-to-Peer Network
- Client/Server Network
- Setup User Accounts
- Provide access to different software packages
- Provide security
- Perform other administrative tasks
- Troubleshoot the network

	Laboratory Activities
Week1 – Week 2	Determining Windows 2000 Compatibility; Installing Server and Workstation; Creating an Answer file; Logon Security Dialogue Box, Using Windows 2000 Help; Customizing Windows 2000 Explorer Embedded Lab Section - Ping Me Embedded Networking Exercise
Week 2- Week 3	Working with Device manager, Adding and Removing Software; Creating Custom MMC Embedded Lab Section - Remote Servo Control Embedded Networking Exercise
Week 4 – Week 5	Installing DNS; Installing a Secondary DNS server; Promoting a Server to a Domain Controller Embedded Lab Section - Evil Embedded Networking Exercise
Week 5 – Week 6	Modifying Logon Locally User Rights; Creating Users and Groups; Account Policies, Creating User Home folders; Local and Roaming user profile; Active Directory Permission Embedded Lab Section - Race for Data Embedded Networking Exercise
Week 7 – Week 8	NTFS Permissions; Adding and Managing Printers; Creating and Connecting to share folders, Creating and Applying Group Policy Objects, Distributing Software using group policy Embedded Lab Section - HTTP Server Embedded Networking Exercise
Week 9 – Week 10	TCP/IP Client Configuration, Installing, Configuring, and Testing DHCP server, Scheduling Task Streaming Video Exercise using RealPlayer or Windows Media Player [3]

Table 1. Lists the main networking activities covered in this course.

Exercises performed with embedded networking nodes will supplement the basic laboratory assignments. The Embedded Node and associated exercises are described in the next section.

Embedded Node and Networking Laboratory Setup

Exercises with Embedded Nodes will demonstrate overhead due to headers and trailers, and encapsulation. The exercises are designed to make the connection between commercial off the shelf equipment such as NIC's, hub's, routers, etc. and the 802.3 data frame. The exercises also are designed so the student can relate the actions of the Embedded Node to a state diagram. In Digital Design and Micro-controller Design courses the EET student is taught to implement state diagrams in logic, VHDL, and C programming if using micro-controllers.

The micro-controller used in the Embedded Node is required to handle communications with a 802.3 transceiver and also accept input and display output. The input and output for the Embedded Node is quite simple. The inputs will consist of analog input from a rheostat, on/off pushbuttons and eight dip switches. The output will be through 1 of 8 LED's or through a D/A port on micro-controller to a speaker or servomotor.

The Embedded Internet nodes, use an EDTP Packet Whacker card (NIC) connected to a Cypress Micro-controller [12]. There are only two embedded nodes, but students can rotated through these as with other network components. The two embedded network nodes cost well under $200 total. The $200 cost does not include the $100 for the Cypress C compiler that works with their free PSoC Designer Environment. The two network nodes are used to send information to one another across the class network. A rheostat on one Embedded Node can control the position of a servomotor on ther other Embedded Node located across the room. As the rheostat dial is rotated the servomotor moves a corresponding number of degrees.

The EDTP Packet Whacker basically is a RTL8019 connected to the necessary magnetics and RJ-45 connector. A set of data, address and control lines connect the RTL8019 to the micro-controller. The full speed of the RTL8019 will not be utilized by the micro-controller, but for the lab exercises it is not necessary that maximum data rates be obtained. Figure 1 depicts the block diagram depicts a single lab embedded node.
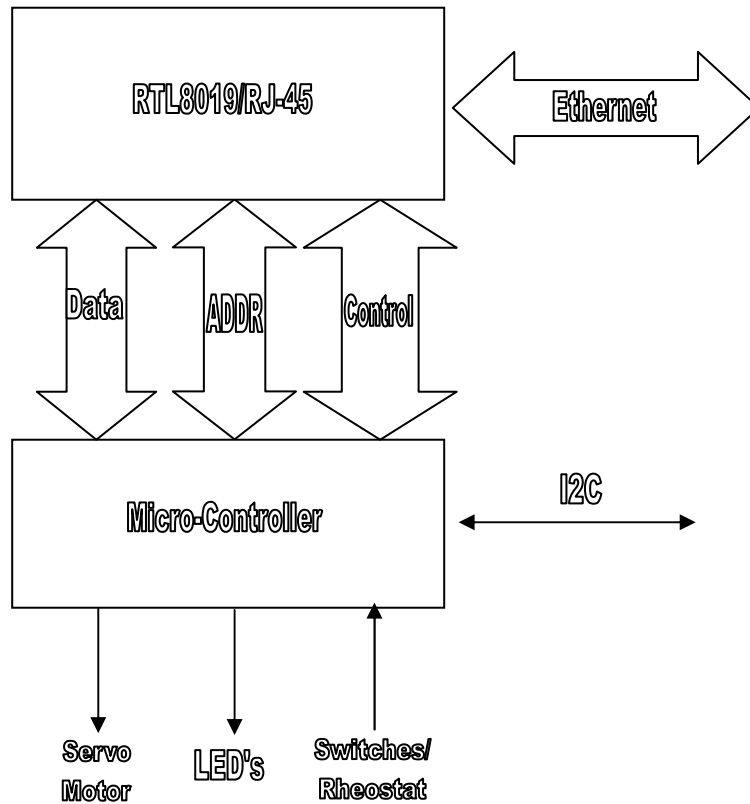
Figure 1. Embedded Node Block Diagram

The two Embedded Nodes will be configured identically, except for the servomotor. There is currently only one servomotor.

The EDTP Packet Whacker core is the RTL8019 Realtek Full-Duplex Ethernet Controller. The format of a Ethernet Frame is as follows [16]:

Preamble	Frame delimiter	Destination Address	Source Address	Length/ Type	Data	Checksum
7 byte	1 byte	6 byte	6 byte	2 byte	46-1500 byte	4 byte

Figure 2. 802.3 Frame

The software that runs in the micro-controller sets the source, designation, length/type field, data and checksum. The Embedded Node with the capability to fail calculating the correct checksum or say put in the wrong Source Address.

Although the basic Ethernet frame can be used to transmit data such as switches, sensors or servo controller commands adding more layers and thus greater overhead allows the embedded controller to function in more complex network environments.

An IP header such as shown in Fig 3 can be encapsulated within the 802.3 frames. The State Machine depicted in Fig 4 of a simple router indicates which bytes in the MAC address are substituted from a list of static MAC addresses based upon the value of the IP address. Once the connection between a network device and a state machine, the EET Student is able to relate the example to other design courses in the program such as Digital Design using FPGA's or micro-controllers. The connection between network devices, and state machines is made during lectures and lab exercises.

Figure 3 shows the actual layout of the IP Header.

```
-----------------------------------------------------------
 Version  |  IHL  |  TOS   |  Total length
 4 Bits   | 4 Bits | 8 Bits |  16 Bits
-----------------------------------------------------------
 Identification            | Flags  | Fragment
 16 Bits                   | 3 Bits | offset 13 Bits
-----------------------------------------------------------
 TTL             |Protocol| Header checksum
 8 Bits          | 8 Bits  | 16 Bits
-----------------------------------------------------------
 Source IP address
 32 Bits
-----------------------------------------------------------
 Destination IP address
 32 Bits
-----------------------------------------------------------
 Options and                     | padding
                                 |
-----------------------------------------------------------
 Data ...
-----------------------------------------------------------
```

Figure 3. IP Header

Figure 3 shows a state machine diagram typical of those used in the course. The state diagram below shows how a very simple static router passes a TCP/IP message using the input IP address to determine the resulting destination MAC address field. Two Embedded nodes connected through I2C can be used to create such a simplistic router.
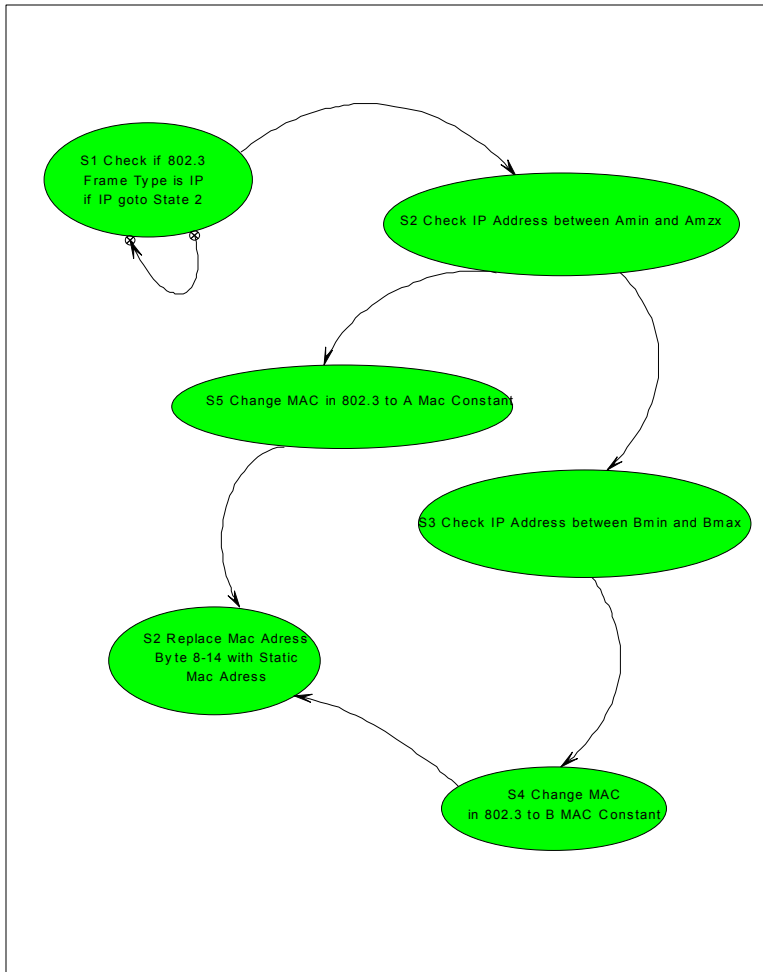
Figure 4. Simple Static Router

The software EDTP Inc. provides with the Packet Whacker to run in the Microchip PIC16F877 micro-controller, supports ARP, Ping, Echo and ICMP (http://www.edtp.com). Embedded Nodes are implemented using the Cypress PSoC micro-controller [13,14,15]. Other advantage's to using the Cypress PSoC are the analog capabilities in the programmable analog array.

The EDTP Inc. software sets the Packet Whacker IP address as a constant. The software with some modification could allow the embedded node to communicate with a PC web (HTTP protocol) browser. Communications using HTTP protocol, increases the overhead due to extra bytes that must be encapsulated within the message.

Included in the lab several exercises to emphasize the trade offs between network layers and message overhead tradeoffs. The exercises will require some understanding of micro-controller programming but the main emphasis will not be on the mechanics of program development or code, etc. Lab exercise emphasis will be to demonstrate with simple examples, using different networking layers to demonstrate communication from simple to complex systems.

The EET Student then has some experience to comprehend similar networking concepts from RS-422 to RapidIO.

Although the basic embedded circuits are simple, the variety of networking concepts that can be demonstrated is not. One embedded node rheostat controls the position of a Servo motor connected to the other embedded node. The two embedded nodes can be directly attached through a cross over connection or connected through a hub or across routers. The micro-controllers could be programmed to pass Dip Switch settings on one embedded node and display the results on the LED's located on the other embedded node. A PC can be used to ping one or both embedded nodes [17,18].

Adding static bytes to the data section allows the browser on the PC to communicate with the embedded node since it would be using HTTP. Another potential use of embedded nodes in the lab network would be to demonstrate an incorrectly working network device or malevolent device. The following is a initial list of lab exercises which at the moment is taken from a subset of the aforementioned potential group of possible embedded node uses:

Exercise Name	Description
Ping Me	PC Pings Embedded Nodes
Remote Servo	Rheostat on one Embedded Node controls Servo Motor on other Embedded Node
Evil Embedded Node	Embedded Node Used to cause errors on network by Misbehaving
Race For Data	How much data do two Embedded Nodes pass by Ethernet and I2C? Data Rates, Data Overhead, Distance Between Nodes will all be features of this exercise.
HTTP Server	Embedded Node acts as Server to PC Browser.

Table 2. Embedded Lab Exercises

The most elementary embedded lab setup would require the two embedded TCP nodes to communicate via a cross over cable [16].

The HTTP Embedded server requires the Embedded node to be connected to a PC. The lab setup for Ping Me also uses the cross over cable to connect the Embedded Node to a PC NIC card with a RJ-45 connector. The following diagram depicts the laboratory setup for Ping Me, or HTTP Embedded Server Exercises:
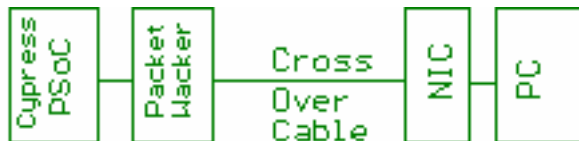

Figure 5. Embedded Lab Setup when using a PC

The lab setup for Remote Servo control uses the cross over cable to connect one Embedded controller with the other. The Evil Embedded Node is meant to run in a wider network environment to demonstrate disruption of a larger network. Race for Data requires both, Embedded nodes to be connected via the cross over cable and two additional connections for the I2C interface [17].

The purpose of the Race for Data is to compare data bandwidth of TCP/IP with an I2C connection. I2C and TCP/IP do not usually compete as technologies. I2C is for short distances and connections between IC's on a PCB or between PCB's. I2C sends data one byte at a time. TCP/IP sends a data packet with a minimum size, which is not efficient for sending a single value from a sensor. The Packet Wacker (NIC) provides a faster link than the fully software implemented bit banging I2C interface. So the implementation of the protocol in software vs hardware elements is also a issue in the Race for Data Exercise.

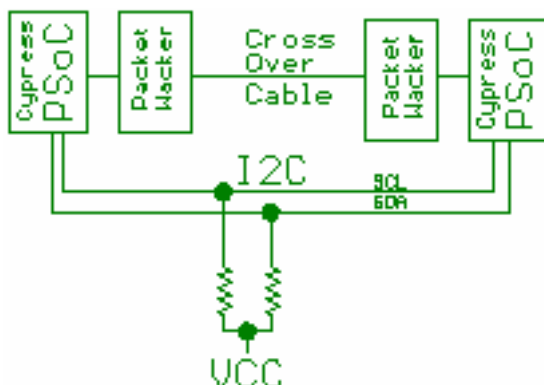The following diagram depicts the Race for Data laboratory setup:

Figure 6. Race for Data Lab Setup

For values of the resistors on the I2C bus see the I2C Master data sheet under Digital COMM blocks in PSoC Designer.

The same assignments may be performed over more TCP/IP complex networks to help emphasize network equipments relationship to the TCP and IP network layers. Several example network configurations that might be used with some of the aforementioned exercises:

Exercise	Network Devices between two Embedded Nodes or Embedded Node and PC	Network Layers Demonstrated
Ping Me	Embedded Nodes connected through Hub	TCP
Remote Servo	Embedded Nodes connected through Hubs and Router	TCP/IP
HTTP Server	Embedded Node connected to PC through cross-over cable	TCP/IP and HTTP

Table 3. Modifications to Lab Exercises

Outcome

Twenty students from Savannah State University and five students Savannah Technical
College completed the course. However, ten students from Savannah State University passed the Windows 2000 Professional (Exam 210) and Windows 2000 Server (Exam
217) and obtained the Microsoft Certified Professional certification. Furthermore, students gained some understanding of embedded networking. Students also gained insight into state machines, that implement networking devices.

References

1. www.can.bosch.com CAN bus developer and owner
2. www.lin-subbus.org LIN bus organization
3. On line material from www.techonline saved presentations:
802.11: Current Trends Oct 16, 2003 Intel
802.16 Broadband Wireless Access The Next Big Thing in Wireless, Intel, Nov 20, 2003
Advanced Switching Architecture and Implementation Update, Intel, Nov 19, 2003
Serial Switched PCI: Expanding Architectural Possibilities, Stargen Dec 17, 2003
I2C Logic – Expanding a Mature Technology, Phillips, Nov 18, 2003
4. Physical Layers and Video Standards: Solving Customer Design Challenge Oct 1, 2003
Cypress
5. Advanced Switching Architecture and Implementation Update Nov 19, 2003 Intel
6. Advanced Switch standard (www.asi-sig.org)
7. www.semiconductors.philips.com Design Con 2003 Tec Forum I2C Bus Overview Jan 27,
2003
8. RTC Sep 2003 pg 34 Advanced – A Modular Open Platform Standard for the Next Decade Joe
Pavlat,
PICMG
9. Circuit Cellar Online, Ethernet Technology Part 1: Frames, Collisions, and 10-Mbps LANs,
James
Antonakos March 2002
10. Circuit Cellar Online, Ethernet Technology Part 2: The 100- and 1000-Mbps Ethernet, James
Antonakos, April 2002
11. Circuit Cellar Online, Ethernet Technology Part 3- Nework Building Blocks, James Antonakos,
April
2002
12. Circuit Cellar Online, Introducing the Packet Whacker Part 1: Hitching a Ride on the
PICEM.NET,
Fred Eady, Oct 2001
13. RTL8019AS Realtek Full-Duplex Ethernet Controller with Plug and Play Function (RealPNP)
Advance
Information Ver 2, Realtek Semi-Conductor CO., LTD. Head Office 1F, NO. 11, Industry E. RD.
IX,
Science-Based Industrial, Park, Hsinchu 30077, TAIWAN, R.O.C. Tel: 886-35-780211 FAX: 886-
35-
776047, 15 Jan 1996
14. RTL8019AS Realtek Full-Duplex Ethernet Controller with Plug and Play Function (RealPNP)
Advance
Information Ver 2, Realtek Semi-Conductor CO., LTD. Head Office 1F, NO. 11, Industry E. RD.
IX,
Science-Based Industrial, Park, Hsinchu 30077, TAIWAN, R.O.C. Tel: 886-35-780211 FAX: 886-
35-
776047, LS003.6 1995.04.25
15. DP8390 Network Interface Controller: An Introductory Guide, National Semiconductor
Application
Note 475, May 1993
16. 802.3® IEEE Standard Information technology – Telecommunications and information
exchange
between systems – Local and metropolitan area networks – Specific requirements Part 3: Carrier
sensse
multiple access with collison detection (CSMA/CD) access method and physical layer
specifications, IEE

STD 802.3®-2002 (Revision of IEEE Std 802.3, 2000 Edition), The Institute of Electrical and Electronics
Engineers, Inc., 3 Park Avenue, New York, NY 10016-5997, USA, 8 March 2002
17. Circuit Cellar, A Prototype Internet Based Security System, Skylar Lei, Jim Haslett, and Michael
Smith, Aug 1999
18. Circuit Cellar, Still Swimming with the STK500 Onto the JTAG ICE, Fred Eady, June 2002

ASAD YOUSUF
Asad Yousuf is a Professor of Electronics Engineering Technology at Savannah State University. He received his BS in Electrical Engineering from the NED Engineering University, Karachi, Pakistan in 1980 and MS in Electrical Engineering from the University of Cincinnati in 1982 and an Ed.D. from the University of Georgia in 1999. Asad is registered as a Professional Engineer in Georgia, and a Microsoft Certified System Engineer (MCSE).

Muniyappa Venkatesha
Muniyappa Venkatesha is a head of the department of Electronics and Computer Engineering Technology at Savannah Technical College. He received his BE degree in Electrical Engineering from BDT college of Engineering Davanagere, Mysore University, India in 1967 and MS in Electrical Engineering from the North Carolina A&T University of Greensboro in 1985 and Ed.S from the Georgia Southern University in
1993.

WILLIAM LEHMAN
William Lehman received his BS in Electrical Engineering from the Catholic University of America, D.C.
in 1979. He has worked through the years testing software and hardware systems in the aerospace and telecommunications industries. He is currently a consultant and may be reached at Advanced Systems & Materials, Lenardtown, Maryland (water@tqci.net).