

Public District School Board Writing Partnership

Course Profile Computer Engineering Technology

Grade 10

Open

• *for teachers by teachers*

This sample course of study was prepared for teachers to use in meeting local classroom needs, as appropriate. This is not a mandated approach to the teaching of the course. It may be used in its entirety, in part, or adapted.

Course Profiles are professional development materials designed to help teachers implement the new Grade 10 secondary school curriculum. These materials were created by writing partnerships of school boards and subject associations. The development of these resources was funded by the Ontario Ministry of Education. This document reflects the views of the developers and not necessarily those of the Ministry. Permission is given to reproduce these materials for any purpose except profit. Teachers are also encouraged to amend, revise, edit, cut, paste, and otherwise adapt this material for educational purposes.

Any references in this document to particular commercial resources, learning materials, equipment, or technology reflect only the opinions of the writers of this sample Course Profile, and do not reflect any official endorsement by the Ministry of Education or by the Partnership of School Boards that supported the production of the document.

© Queen's Printer for Ontario, 2000

Acknowledgments

This profile is the result of a collaborative effort between the Simcoe County District School Board and the Institute for Catholic Education (ICE).

Public School Board Writing Team - Grade 10 Computer Engineering
Lead Board

Simcoe County District School Board
Robert Emptage, Laura Featherstone, Project Managers

Course Profile Writing Team - Public

Dave Zdyrko, Renfrew County Board of Education (District #28), Lead Writer
Ann Pepin, Simcoe County District School Board (editor/writer)

Catholic School Board Writing Team - Grade 10 Computer Engineering

Lead Board

Dufferin-Peel Catholic District School Board
Denise Panunte, Project Manager

Course Profile Writing Team - Catholic

Peter Fujiwara, Dufferin-Peel Catholic District School Board, Lead Writer
Gerard Morris, Dufferin-Peel Catholic District School Board
Graham Smyth, Chatham Catholic District School Board (retired)

Course Overview

Computer Engineering Technology, Grade 10, Open

Identifying Information

Development Date: 1999

Course Title: Computer Engineering Technology

Grade: 10

Course Type: Open

Ministry Course Code: TEE20

Credit Value: One

Description/Rationale

Throughout this course, students examine computer hardware and the control of external components from an engineering perspective. Students solve problems and study the functions of key computer components and peripherals, logic gates, fundamental programming concepts, internal numbering and character representation systems, operating systems, and networks. They also develop an awareness of future educational opportunities and careers in the field of computer engineering. This course is designated as open and can be taken by all students who wish to learn about Computer Engineering Technology. Students who wish to continue study in this area can take the Workplace or University/College courses in Grades 11 and 12 that lead to post-secondary courses or entry into the workplace.

Unit Titles (Time + Sequence)

Unit 1	Computer Engineering Hardware	13 hours
Unit 2	Integrated Circuits	20 hours
Unit 3	Networking	20 hours
Unit 4	Computer Programming	27 hours
Unit 5	Computer Interfacing	30 hours

Unit Descriptions

Unit 1: Computer Engineering Hardware

Time: 13 hours

Description

In this unit, students identify and explain the functions of the basic components of a computer, basic circuits, and peripheral devices. Emphasis is placed on safety as students handle a variety of tools, equipment, and internal and external components. Students create a simple circuit, glossary of terms, database of lab components and individual computer log sheets for recording upgrades or changes. Students also identify employability skills and explore careers in the computer industry.

Unit 2: Integrated Circuits

Time: 20 hours

Description

The focus of this unit is on integrated circuits and how the internal workings of a computer represent data such as characters and numbers. Students learn standard codes for internal numbering and character representation. They learn to design and construct fundamental logic gates (i.e., AND, OR, NOR, NAND, NOT, XOR, XNOR). They also learn about and construct simple electronic circuits, apply Boolean algebra, and devise truth tables to test and describe their functionality. Students develop an understanding of gates, semi-conductors (e.g., transistors, diodes, etc.), and integrated circuits by designing and building simple logic gates.

Unit 3: Networking

Time: 20 hours

Description

Students explore and set up parallel and series computer communication processes within a computer and between computer systems (e.g., internal architecture, cabling standards, topology, and network types). They use problem-solving skills to apply their knowledge to tasks such as researching simple network types and building simple communication networks. Students also learn about the importance of network connectivity and infrastructure and how it impacts on our world as well as potential career opportunities in the area of computer networking.

Unit 4: Computer Programming

Time: 27 hours

Description

This unit focusses on how to program a computer using a problem-solving model. This model helps to organize and develop the fundamental structures of programming. These fundamental structures include variable declarations, assignment statements, input/output, selection, and looping. Each structure builds upon and is incorporated into subsequent structures. The programming software introduced in this unit allows students to write simple programs which integrates with hardware to control external devices and peripherals. Students also research and identify computer-related careers and explore ergonomics.

Unit 5: Computer Interfacing

Time: 30 hours

Description

The final culminating unit incorporates information learned in all previous units. Students are expected to work through Activities 1, 2, and 3 to fully prepare them for the challenge of Activity 4 in which they complete a project that demonstrates their knowledge of integrating software and hardware processes to solve an interfacing challenge.

Course Notes

Teachers must ensure students have paper or electronic copies of expectations being assessed/evaluated in each activity. As students become more familiar with rubrics and checklists, they can be involved in adapting and designing assessment tools. This gives students more ownership of their learning goals and greater self-direction in their learning. In every unit students work in a hands-on environment and must have clear goals to successfully meet course expectations. Teachers must continually conference with students to ensure they understand their progress to date and to suggest areas needed for improvement.

The use of a daily log or journal is a useful tool for students to practise their writing skills and technology terminology. It may be used for on going activities as a terminology database, computer career database, computer terminal log, and/or creation of a student portfolio for exemplary work. Students have various opportunities to practise time-management skills and follow printed instructions in both individual and group work settings.

The use of hardware and software resources must be planned to ensure students have access to basic hand tools, materials, and equipment for designing and constructing circuits and to access a small computer network that is not networked into the main school system. They also require computers that are part of the main system for research and software application tasks. The facilities must allow students to disassemble, design, and assemble a variety of basic electronic circuits and computer hardware systems. Connections to the community are an ideal source of used equipment to serve many functions. Teachers must review school board policies involving appropriate student use and access to Internet services. See the Grade 10 Computer and Information Science course profile for activities to assist students in using the Internet.

Teaching/Learning Strategies

All units are activity-based. Socratic lessons, teacher demonstrations, and research activities provide students with the necessary terminology and methodology necessary to complete the various activities. Classroom discussions, brainstorming, and collaborative and/or co-operative learning is used to assist students in meeting course expectations. Problem-solving exercises are used. Students also research, write reports, and take notes in meeting expectations. Upon completion of this course, students demonstrate the ability to apply skills and knowledge to practical work tasks that involve planning and implementation processes, completion of work assignments, and various problem-solving activities.

Assessment/Evaluation Techniques

Diagnostic testing may be incorporated at the beginning of the units for teachers to assess the knowledge variance and experiences of students in their classes.

The teacher assesses/evaluates students in a formative manner by using roving student conferences where the focus can be on skills and knowledge, teamwork, co-operative learning, etc. Checklists are used to assess the operational steps of a process. Checklists and step-by-step instructions vary depending on the resources available at individual schools. Self-evaluation may also be used to help students develop a sense of responsibility for their own learning. Teacher-student conferencing provides clarity, maintains the expected standards, and assists students in defining steps they need to take for improvement. Periodic assessment of the student portfolio also assists the teacher in providing students with formative assessment. The addition of a self-and peer assessment, in group work situations, helps to identify students and/or groups who need support. Performance tests are effective for assessing the achievement of knowledge and skills. The vocabulary used in the test questions should reflect that used in the lab situation. Although students should be encouraged to write answers in proper sentence form, questions and answers that involve diagrams can be an effective assessment instrument. Final evaluation may include the last activity in Unit 5, the student portfolio, and glossary of terms.

Suggested Student Assessment/Evaluation Table

Course Grade Weighting	%
Application Exercises/Problem solving	40
Tests	30
Assignment/Presentation (Final evaluation)	30

Accommodation

Teachers using this course profile are expected to be acquainted with the students' Individual Education Plans (IEPs) and their unique learning characteristics, and to make the necessary accommodations.

There are a wide range of teaching/learning strategies used to ensure that the needs of all students are met. Written tests should be designed to suit the reading and writing levels of the students. Teachers are encouraged to modify and expand teaching strategies to accommodate learning styles. These may include:

- modified approaches to evaluation;
- option for oral testing and student demonstrations of acquired skills;
- student-to-student discussion and teacher-to-student discussion to encourage confidence and motivation;
- small group learning;
- flexible timelines;
- adaptation of handouts;
- peer tutoring;
- enrichment and extension activities.

Resources

This list contains the most common resources and each activity has additional specific resources listed.

Human Resources

Community libraries and School Library/Resource Centre

School board technical service department personnel

Community partners and computer industry personnel

Software

Operating systems (e.g., DOS, Windows 9x, Windows NT, System 6.x, 7.x, 8.x, Unix, or others)

OESS software tools (e.g., Corel *WordPerfect*, *Microsoft Works*, *Appleworks*, etc.)

Web and FTP Server and client applications

Print

Baker, J. *Digital Computer Technology: An Introduction*. Toronto: McGraw-Hill Ryerson, 1983.
ISBN 0-13-211947-1

Blissmer, Robert H. *Introducing Computers*. New York: John Wiley & Sons, Inc., 1991.
ISBN 0471-53443-9

Feldman, Jonathon. *Sams Teach Yourself Network Troubleshooting in 24 Hours*. Sams, 1998.
ISBN 0672314886

Gregg, Kenneth. *Windows Networking Basics*. Harper Collins Canada, 1998. ISBN 0764532146

Kearns, Dave. *Sams Teach Yourself Windows Networking in 24 Hours*. Sams, 1998.
ISBN 0672314754

Keogh, Jim. *Core MCSE: Networking Essentials*. Prentice-Hall of Canada Ltd.
ISBN 0130107336

Lawrence, Orville. *Computer Technology*. Toronto: McGraw-Hill Ryerson, 1984.
ISBN 0-07-548711-X

Magendanz, Thomas and Popescu-Zeletin. *Intelligent Networks: Basic Technology, Standards & Evolution*. International Thomson Press, 1996. ISBN 1850322937

Network A+ Certification Study Guide. Syngress Media, Inc., 1999. ISBN 0-07-211846-6

MCSE Networking Essentials For Dummies, Training Kit. IDG Books Worldwide, 1999. ISBN 0764506218

Norton, Peter. *Essential Concepts*. McGraw-Hill Ryerson Limited, 1999. ISBN 0-02-804394-4
Operating System Manuals and reference texts

Parsons, Oja. *Computer Concepts*. 1996. ISBN 0-7600-3440-0

Shelly, Gary and Thomas Cashman. *Computer Fundamentals for an Information Age*. California: Anaheim Publishing Co., 1984. ISBN 0-88236-125-2

Simms, Forrest. *Getting Started in Electronics*. USA: Radio Shack, 1983. Cat No 276-5003

White, Ron. *How Computers Work*. Quebec, Canada: 1997. ISBN 01-56-276546-9

**The Whole Internet: The Next Generation*. O'Reilly and Associates Inc., 1999. ISBN 1-56592-428-2

Web Sites

Novell Network Primer – network solutions, questions, and links
<http://www.novell.com/catalog/primer/primer.html>

IT Careers – careers database
<http://www.itcareers.com>

Intel Resources - product listing, resources, and links
<http://www.intel.com/education/k12/resources/index.htm>

Cisco Certification CCIE – program requirements and information.
<http://www.cisco.com/warp/public/625/ccie/>

Microsoft Educational Resources
<http://www.microsoft.com/education/instruction/default.asp>

3Com's Netprep programme – educational information and links
<http://education.3com.com/Netprep/index.html>

Motorola Processors and Micro-controllers – product listings and resources
<http://www.motorola.com>

Video

Intel Corporation. *The Journey Inside*. Part of *The Journey Inside Education* kit.
<http://secure.wesweb.com/intel/form.htm>
Contains two videos, an instructional binder, and electronic components

OSS Policy Applications

The Grade 10 Computer Engineering Technology course is designated as a Technological Education program. All courses offered in technological education at the Grade 10 level are open courses, which comprise a set of expectations that are appropriate for all students. (See *The Ontario Curriculum, Grades 9 and 10, Program Planning and Assessment, 1999* for a description of the different types of secondary school courses.) Students can use the course as a compulsory credit (1 credit from Science [Grade 11 or Grade 12] or Technological Education [Grades 9 –12]), or as an additional credit. This course is designed to provide students with a broad educational base that prepares them for their studies in Grades 11 and 12, post-secondary education, and entry into the workplace. Anti-discrimination education, equity/social justice issues, career goals/co-operative education, and community partnerships are also discussed in this course. All of these support many of the Ontario Secondary School Policies.

Career exploration throughout all units is available to students with specific reference to *Choices into Action: Guidance and Career Education Program Policy for Elementary and Secondary Schools, 1999*.

Course Evaluation

Teachers may evaluate their course through a variety of methods. Teachers may network with colleagues from other schools, subject associations, and peers at the local school to determine what modifications or new ideas could be incorporated into the units. Since every teacher approaches the units in a unique way, there are ample opportunities for extensions, modifications and applications. The school and business community may have input on aspects of the course.

The following areas are assessed:

- Expectations are being met.
- The learning styles of all students are being considered and reflected through teaching strategies.
- Assessment/evaluation techniques measure student expectations in a reliable and accurate manner.
- Parents are informed of student performance on a regular basis.
- Variety of assessment/evaluation tools are used.
- Variety of teaching/learning strategies are used.
- Special needs of individual students are being met.
- Community involvement is incorporated into the course.

Have students complete the following checklist. (The criteria for the evaluation chart is as follows: 1 = never, 2 = sometimes, 3 = frequently, 4 = always and with a variety of strategies.)

Computer Engineering Technology Course Evaluation

1. The course covered all the topics and expectations given at the beginning of the semester.	1	2	3	4
2. There were opportunities to learn all the concepts.	1	2	3	4
3. The activities were relevant and interesting.	1	2	3	4
4. The methods of assessment/evaluation were valuable to me.	1	2	3	4
5. There were references, visits, and connections to the computer engineering industry.	1	2	3	4
6. Reporting comments accurately reflected my progress.	1	2	3	4
7. Overall rating of this course	1	2	3	4
Additional Comments: (You may wish to comment on an activity that you particularly enjoyed, an activity you would like to see changed, and/or thoughts you have on the course.)				

Coded Expectations, Computer and Engineering Technology, TEE2O

Theory and Foundation

Overall Expectations

TFV.01E

– describe how the internal components of the computer enable the peripherals to function;

TFV.02E

– describe a problem-solving model such as the input, processing, output model;

TFV.03E

– explain internal numbering and character representation systems;

TFV.04E

– describe and illustrate the functions of logic gates;

TFV.05E

– describe the fundamental programming constructs.

Specific Expectations

Computer Logic

TF1.01E

– describe the relationship between the binary number system and computer logic;

TF1.02E

– define a standard way of representing characters in binary code;

TF1.03E

– describe the function of the fundamental logic gates, including the function of each pin: AND, NAND, OR, NOR, XOR, XNOR, and NOT.

Hardware, Interfaces, and Networking Systems

TF2.01E

– use precise terminology in relation to all hardware, interfaces, and networking systems;

TF2.02E

– identify the basic internal and external components of a computer;

TF2.03E

– describe the primary function of each basic component;

TF2.04E

– identify computer internals and peripheral devices and describe their relationship.

Programming Concepts

TF3.01E

– define constants, variables, expressions, and assignment statements, including the order in which the operations are performed;

TF3.02E

– describe how computers store and work with different types of data, including numbers and characters.

Skills and Processes

Overall Expectations

SPV.01E

- connect and use correctly a variety of computer components and peripherals;

SPV.02E

- demonstrate the use of an operating system, including a network;

SPV.03E

- use logic gates to construct simple circuits;

SPV.04E

- apply fundamental programming constructs to develop programs that interact with external components.

Specific Expectations

Computer Logic

SP1.01E

- convert positive integer numbers to binary form;

SP1.02E

- derive the truth tables of the fundamental logic gates;

SP1.03E

- write Boolean equations for the fundamental logic gates.

Hardware, Interfaces, and Networking Systems

SP2.01E

- set up a desktop computer system and install software;

SP2.02E

- build an interface to connect the computer to a simple peripheral device;

SP2.03E

- trace the operation of a system consisting of a program, an interface, hardware, and directories;

SP2.04E

- use appropriate file management techniques;

SP2.05E

- use correctly a variety of network system software;

SP2.06E

- use Internetworking services correctly to access and navigate global information resources.

Programming Concepts

SP3.01E

- use input and output statements in a program;

SP3.02E

- use a decision structure and a repetition structure in a program;

SP3.03E

- design, write, and test a computer program to control a simple peripheral device.

Impact and Consequences

Overall Expectations

ICV.01E

– describe the evolution of computer electronics;

ICV.02E

– identify the social impact of computers and associated technologies;

ICV.03E

– identify related computer careers.

Specific Expectations

IC1.01E

– use appropriate strategies to avoid potential health and safety problems associated with computer use, such as posture problems, eye strain, and musculoskeletal injuries;

IC1.02E

– use safe practices in the handling of computer hardware and electronic components;

IC1.03E

– identify important scientific advances in computer electronic components;

IC1.04E

– describe the development of computer engineering technology and its impact;

IC1.05E

– describe careers related to computer engineering;

IC1.06E

– analyse the influences of computers on the engineering profession;

IC1.07E

– describe how computer engineering has evolved and how it has affected people's security, safety, and privacy;

IC1.08E

– demonstrate understanding of the importance of ethical computer use;

IC1.09E

– demonstrate compliance with acceptable-use policies;

IC1.10E

– identify computer skills that are important to employers.

Unit 1: Computer Engineering Hardware

Time: 13 hours

Unit Description

In this unit students identify and explain the functions of the basic components of a computer, including its internal components and peripheral devices. Emphasis is placed on safety as students handle a variety of internal and external components. Students create a database of lab components and individual computer log sheets for recording upgrades or changes. Students develop an awareness of the impact and consequences of computer use in the world and the impact of technology on the value structures of today's society. Students also identify employability skills and explore careers in the computer industry.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations: TFV.01E, TFV.02E, SPV.02E, ICV.01E.

Specific Expectations: TF2.01E, TF2.02E, TF2.03E, TF2.04E, IC1.02E, IC1.09E, IC1.10E, SP2.01E, SP2.04E.

Activity Titles (Time + Sequence)

Activity 1	Computer Engineering Hardware	210 minutes
Activity 2	Operating Systems	150 minutes
Activity 3	Data Storage and Management	75 minutes
Activity 4	Nomenclature	225 minutes
Activity 5	Skills and Career Opportunities	120 minutes

Prior Knowledge Required

- Basic keyboarding skills.
- Application software including word-processing and database software.

Unit Planning Notes

Consider some of the following issues when organizing the facility/lab:

- ratio of computers to students;
- pairs/groups or stations set-up around the room;
- preparation of demonstrations prior to a lesson and test for functionality;
- checking equipment prior to major hands-on activities, perhaps using students;
- spare parts;
- display safety procedures on bulletin boards and refer to them regularly.

Teaching/Learning Strategies

- This unit requires teacher demonstrations (including videos if available) and many hands-on activities for the students. Safety must be addressed at the beginning of the unit and considered throughout each activity. If limited resources are available, various learning stations may be set up to support activities. Other teaching and learning strategies include class discussion, brainstorming, Socratic lessons, peer and teacher conferencing, research and, activity instruction guides.
- Teachers must reinforce with the students that there are many types of computer systems.

Assessment and Evaluation

Some students may possess a considerable amount of prior computer knowledge and skills while others may have very little prior experience. A mixture of assessment strategies (both formative and summative) must be used including assignments, skill demonstrations, quizzes, tests, self-assessment, and roving conferences.

Resources

Student Services and Co-op Departments

Employability Skills Training Brochure, developed by the Conference Board of Canada

Software

Operating systems (e.g., DOS, Windows 9x, Windows NT, System 6.x, 7.x, 8.x, Unix or others)

OESS software tools (e.g., Corel *WordPerfect*, *Microsoft Works*, *Appleworks*, etc.)

Web and FTP Server and client applications

Print

A+ Certification Study Guide. Syngress Media, Inc., 1999.

Blissmer, Robert H. *Introducing Computers*. New York: John Wiley & Sons, Inc., 1991.

ISBN 0471-53443-9

Norton, Peter. *Essential Concepts*. McGraw-Hill Ryerson Limited, 1999. ISBN 0-02-804394-4

Operating System Manuals and reference texts

Parsons, Oja. *Computer Concepts*. 1996. ISBN 0-7600-3440-0

White, Ron. *How Computers Work*. Quebec, Canada: 1997. ISBN 01-56-276546-9

The Whole Internet: The Next Generation. O'Reilly and Associates Inc., 1999.

ISBN 1-56592-428-2

Web Sites

Canada WorkinfoNet – source of career, education, and labour market information for Canadians

<http://www.workinfo.net.ca>

Canadian Association of Professional Engineers – education and careers

<http://www.apegga.com/>

Canlearn Interactive: Human Resource Development Canada – career information

<http://www.canlearn.ca>

Human Resource Development Canada – career information

<http://www.hrdc-drhc.gc.ca/maps/national/canada.shtml>

Information by National Occupation Classification Code

<http://www.eoa-hrdc.com/3519/menu/occnoc.stm>

Information Technology Careers – career search information

<http://www.itcareers.com>

Intel – processors and micro-controller information

<http://www.intel.com>

Motorola – Motorola processors and micro-controllers information

<http://www.motorola.com>

Novell – networking primer with links and a glossary

<http://www.novell.com/catalog/primer/primer.html>

Ontario Association of Certified Technicians and Technologists – education and careers

<http://oacett.org/>

The Toronto Star Newspaper

<http://www.thestar.com/>

The Toronto Sun Newspaper

<http://www.canoe.ca/TorontoSun/home.html>

Activity 1: Computer Engineering Hardware

Time: 210 minutes

Description

Students are introduced to the historical development of a computer system as well as the relationship between computer internal hardware and software and external devices. Students identify internal components and external devices and peripherals and describe how they interact to facilitate a variety of hardware and software processes. Students also apply acceptable computer use policies and technical facility safety practices.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Impact and Consequences

Overall Expectations

TFV.02E - describe a problem-solving model such as the input, processing, output model;

ICV.01E - describe the evolution of computer electronics.

Specific Expectations

TF2.02E - identify the basic internal and external components of a computer;

TF2.03E - describe the primary function of each basic component;

TF2.04E - identify computer internals and peripheral devices and describe their relationship;

IC1.02E - use safe practices in the handling of computer hardware and electronic components;

IC1.09E - demonstrate compliance with acceptable-use policies.

Planning Notes

- Prepare copies of the Acceptable Use Policy.
- Decide on the format of the student Portfolio (e.g., file folders, binders, duo tangs, electronic storage, or diskettes).
- Prepare notes to accompany a brief history of the computer (Appendix 1.1.3 – History of Computers Outline). The Journey Inside the Computer is the resource for this activity (see Resources).
- Prepare an overhead of the computer model (see Appendix 1.1.4 – The Computer Model).
- Assemble available hardware for hands-on or demonstration activities for identification of internal and external devices. Where hardware is not available, consult Resources to create appropriate overheads, multimedia presentations, and handouts or invite guests from local industry.
- Determine the most effective use of existing hardware within the recommended time allotment (e.g., two to three students per computer, groups of four to six students for connectivity exercises, etc.).
- Test equipment for demonstrations and hands-on activities (e.g., computer storage systems, electrical, etc.). Use appropriately trained students to assist in the set-up.
- Safety with electricity and proper handling of computer components must be addressed.
- Consider stations and having students rotate through various areas to allow for the best use of time and resources.

Prior Knowledge Required

- The computer keyboard and mouse
- Grade 9 Science Unit on Electricity

Teaching/ Learning Strategies

- Introduce the Acceptable Use Policy to the students to put in their student Portfolio for easy reference. Ask students to complete the computer survey questionnaire to hand in (Appendix 1.1.1 – Computer Questionnaire).
- Discuss important safety considerations and begin the first lesson on general classroom safety. Issues such as tying back hair, removal of jewelry and coats, proper eye wear, safety with electricity, and proper handling of computer components must be addressed, as well as basic hand tool safety.
- Introduce students to the student Portfolio as a resource for them as they progress through the course. The use of a Safety Passport may be used for this and all other safety lessons (Appendix 1.1.2 – Sample Safety Passport).
- Show the video *The Journey Inside The Computer*, segment 1 (8 minutes) for an overview of the history of computers. Students complete the History of Computers worksheet (either during or after the video) (Appendix 1.1.3 – History of Computers).
- Show the overhead of the computer model (Appendix 1.1.4 – The Computer Model), giving students definitions for input, information processing, and output to write in their notes. Ask students in pairs to come up with a list of items from their own experience that might fit under the categories of input device or output device. Ask each pair to share their list with the class as the teacher compiles the information for the class to view. Each pair should only mention new items. Teachers should clarify any misconceptions as this activity progresses. Establish these items as external devices.
- Review safety procedures for the handling of basic electronic components with students. Also review basic electricity learned in Grade 9 (e.g., circuit types and the flow of electrical energy).
- Students use low voltage batteries, wires, and lights or buzzers to construct simple, series and parallel circuits as a review of Grade 9 electricity unit. The teacher and students review the schematic symbols for the components in simple electrical circuits (i.e., conductor, source, load, and switch). For homework, students draw schematics and label the circuits they constructed.
- Students inventory the outside of the computer case to identify as many parts as they can (e.g., on/off switch, CD-ROM drive cut-out, disk drive opening, power cord, various ports). Hand out computer terms worksheet (Appendix 1.1.5 - Key Computer Terms) and, using a demonstration model, give students definitions for the terms that apply. Students continually update their glossary of terms and store in their portfolios. Introduce the definition of an integrated circuit (IC): a complex circuit which is constructed in a condensed form into a single electronic component. Note: programmable ICs may run a limited function computer like a watch, however, personal computers (PC) require many ICs to perform many different functions. Specialized computers used in industry such as Programmable Logic Controllers (PLCs) have either very few or a great number of ICs depending on the function they perform.
- Demonstrate, using components from *The Journey Inside* Intel kit or an actual computer, the removal of the system case and give students a tour of the internal devices (e.g., motherboard, CPU, memory, power supply, drives, audio, video, ICs, CMOS battery, and cabling). Students complete the Key Computer Terms worksheet.
- If time and resources allow, have students remove cases from computers to see and touch components.
- End this activity with a knowledge and understanding quiz (Appendix 1.1.6 – Key Computer Engineering Terms and History Quiz).

Assessment/Evaluation Techniques

It is important that students have a good understanding of the basic components of a basic computer system as this activity is the building block of all other activities. In order to ensure this understanding, teachers:

- review the student computer questionnaire to give an idea of computer expertise in the class;
- review with students the circuit drawings they created for homework for accuracy and neatness and provide feedback;
- use the checklist for completed tasks found in student Portfolio (Appendix 1.1.7 – Activity Checklist) (used throughout the unit);
- apply knowledge and understanding quizzes (Appendix 1.1.6 – Key Computer Engineering Terms and History Quiz);
- use learning skills checklist throughout the activity (Appendix 1.1.8 – Learning Skills Checklist).

Accommodations

- Use grouping as a means of allowing students to develop enhanced work and/or for peer mentoring.
- Use visual aids and demonstrations to assist students as needed.
- Provide dictionaries or textbook glossary references for students.
- Use overheads from Intel kit or other resources to supplement discussions of internal and external devices.

Resources

Print

Blissmer, Robert H. *Introducing Computers*. New York: John Wiley & Sons, Inc., 1991. ISBN 0471-53443-9

Norton, Peter. *Essential Concepts*. McGraw-Hill Ryerson Limited, 1999. ISBN 0-02-804394-4
Operating System Manuals and reference texts

White, Ron. *How Computers Work*. Quebec: Ziff Davis Press, 1997. ISBN 01-56-276546-9

Video

Intel Corp. *The Journey Inside The Computer*. (training kit and video)

Web Sites

Novell – networking primer with glossary
<http://www.novell.com/catalog/primer/primer.html>

Appendix 1.1.1

Computer Questionnaire

Name:

Date:

Your answers to the following questions will help me understand the computer background you bring to this course. Please be as specific as possible.

1. Do you have a computer system at home? If no, skip to question #4. If yes, please describe your main system.
2. How much time per week (on average) do you use your computer? For what purpose(s)?
3. Describe a time when something went wrong and you were able to fix the problem.
4. Describe any work you have done using electronic components other than on PCs.
5. Have you ever used computers in other locations? (e.g., Library/Resource Centre, friend's house) For what purpose?
6. List any places you know where integrated circuits (ICs) are used other than in PCs.

Appendix 1.1.2

Sample Safety Passport

This is a sample of a generic safety passport that may be adopted for use in a number of technology classrooms. The purpose for the safety passport is to ensure that students are fully aware of all safety features on each piece of equipment in the technical facility prior to using them independently. This process may be adapted to suit the individual teacher's and students' needs.

The general process is as follows:

1. When the teacher introduces a new piece of equipment, (devices, wiring), the student records the date of the safety demonstration on their safety passport and it is initialled by the teacher (see sample below). While the teacher demonstrates techniques for the safe operation of the machine and personal protective equipment (e.g., proper eye protection, secure loose hair, remove jewellery, protective clothing, etc.), students carefully note the techniques in their notebooks. The teacher also notes attendance; any students who are absent for the safety lesson must be provided with a makeup lesson.
2. Each student must complete a written (or oral) test on the safe operation of the machine tool, outlining all safety features to be observed. Also, all written tests must be kept by the students in their notebooks. These individual machine tests are designed to complement any general facility safety rules. Upon satisfactory completion of the test the student dates the "tested" column and the teacher initials this as complete.
3. Students must demonstrate to the teacher that they have a thorough knowledge of the safety rules for the equipment and are able to demonstrate their competency on the equipment. Once the teacher has observed the required safe set-up and operation of the equipment by a student, the teacher signs off that portion of the student's safety passport.
4. After the student has completed the first three steps, the teacher signs the final column of the safety passport to indicate that the student is able to use that equipment. Students must be able to provide the teacher with their signed passport each time they wish to use a given piece of equipment. A summary document of all the various permissions may be created by the student and signed by the teacher (as permissions are earned); these summary safety passports may be protected with page protectors or laminated.

Sample Safety Passport

Student Name: _____							
Equipment: _____							
See notebook for the notes on safe setup and operation of the equipment.							
Attended Teacher Safety Instruction and Demonstration (and notes recorded)		Passed Written or Oral Testing		Demonstrated Safe Setup and Operation of Equipment to Teacher		Granted Permission to use Equipment by Teacher	
Date of Lesson	Teacher Initial	Date Tested	Teacher Initial	Date of Demo	Teacher Initial	Date	Teacher Initial

Appendix 1.1.3

History of Computers Outline

The First Computers - Abacus

European Calculating Machines

The Jacquard Loom

Charles Babbage

Herman Hollerith

Early Electronic Computers

First-Generation Computers (1951- 1959)

Second-Generation Computers (1959 - 1963)

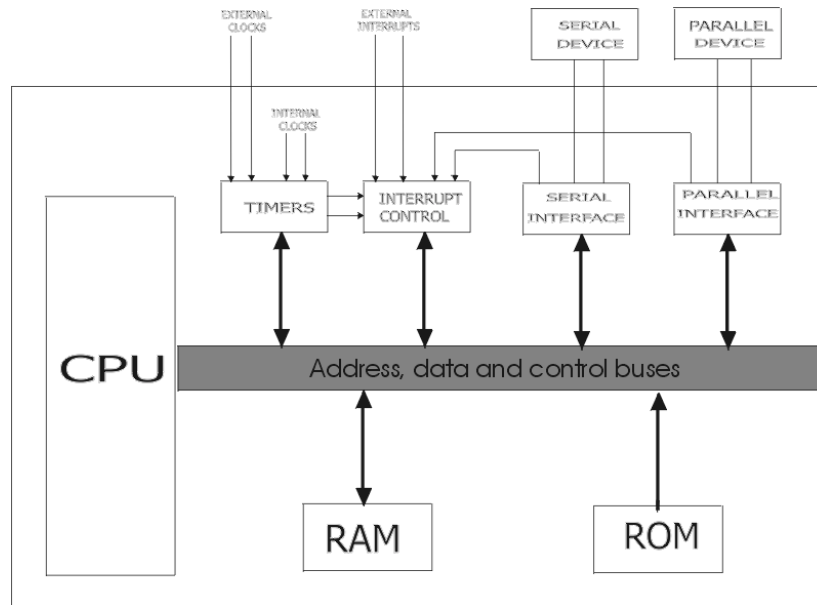
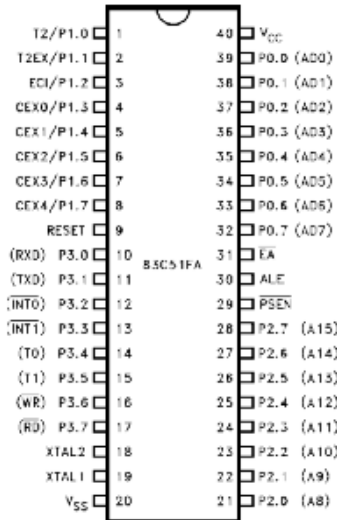
Third-Generation Computers (1963 - 1975)

Fourth-Generation Computers (1975 - present day)

Fifth-Generation (Artificial Intelligence)

Appendix 1.1.4

The Computer Model



Definitions

Input

Information Processing

Output

Appendix 1.1.5

Key Computer Terms

Central Processing Unit (CPU)

BIOS ROM

SIMMs Ram

Cache Ram

ISA

PCI Slot

Power Supply

Floppy Drive

Hard Drive

CD-ROM Drive

Parallel Port

Serial Port

Graphics Port

USB Port

Com2 Port

Keyboard Port

IDE Cable

CMOS Battery

CMOS Ram (Flash Ram)

Integrated Circuit

Micro-controller

PLC

Series Circuit

Parallel Circuit

Switch

Load

Conductor

Computer Cable

Appendix 1.1.6

Key Computer Engineering Terms and History Quiz

Name:

Date:

1. Describe the primary function of five external devices.
2. Describe the primary function of five internal devices.
3. If I was working on a computer system and was having difficulty with output, name two pieces of hardware I could specifically check. From the information learned so far, what would a simple check include?
4. Briefly identify the five generations of computers, explaining the most important feature(s) of each one.
5. Describe where five micro-controllers are used and the function of a PLC.

Answer Key and Suggested Marking Scheme

Questions 1 and 2 - from glossary (10 marks each)

Question 3 - most common answers, monitor and printer. Check cables, power source, reboot system, error messages (5 marks)

Question 4 - from note given (10 marks)

Question 5 – from class discussion (5 marks)

Appendix 1.1.9

Learning Skills Rubric

Expectations	Level 1 (50-59%)	Level 2 (60-69%)	Level 3 (70-79%)	Level 4 (80-100%)
Works Independently	- rarely meets deadlines - rarely focussed	- sometimes meets deadlines - needs reminders to stay on task	- usually meets deadlines - occasionally needs reminders to stay on task	- always meets deadlines - focussed
Teamwork	- rarely tries an alternate role in group work - rarely shares ideas - rarely listens to others	- sometimes tries an alternate role in group work - sometimes shares ideas - listens to others some of the time	- frequently tries alternate role in group work - usually shares ideas - listens to others most of the time	- takes on a variety of roles in group work - always shares ideas - listens to others
Organization	- rarely prepared for class - demonstrates limited organization - rarely uses available resources	- generally prepared for class - demonstrates some organization - uses some of the available resources	- usually prepared for class - demonstrates considerable organization - makes good use of available resources	- always prepared for class - demonstrates effective organization - seeks out and uses additional resources and uses those provided
Work Habits/ Homework	- rarely completes homework - rarely makes up missed work - rarely makes good use of class time	- completes more than half of the homework - makes up some missed work - sometimes makes good use of class time	- completes most homework - makes up most missed work - usually makes good use of class time	- completes all homework - makes up all missed work - always makes good use of class time
Initiative	- rarely acts to solve problems - rarely seeks available assistance	- sometimes acts to solve problems - sometimes seeks available assistance	- usually acts to solve problems - usually seeks available assistance	- always acts to solve problems - always seeks available assistance

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Activity 2: Operating Systems

Time: 150 minutes

Description

Students are introduced to the function of an operating system and the relationship to hardware and application software. Emphasis is placed on understanding of the internal components and how they relate to peripheral devices. Students demonstrate their understanding by using an operating system, including a network, to complete a simple task. Students also research possible future directions of computer operating systems.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes

Overall Expectations

TFV.01E - describe how the internal components of the computer enable the peripherals to function;
SPV.02E - demonstrate the use of an operating system, including a network.

Specific Expectations

TF2.03E - describe the primary function of each basic component;
SP2.06E - use Internetworking services correctly to access and navigate global information resources.

Planning Notes

This activity is designed to help the student understand the need for user-friendly software. The software layers are Application Layer (e.g., word processors, data bases, games which use higher level programming languages). An Operating System Layer (e.g., MS-DOS, Windows 95/NT, Linux, MacOS) can be modified using a lower level programming language (e.g., BIOS: Basic Input Output System) as in changing the software configuration of newly installed hardware components in a system. It would be beneficial to have a computer network, (older computers are fine), to allow the students an opportunity to work with the components. The teacher:

- prepares a presentation of current operating systems, including their school network;
- assembles the hardware for hands-on activities or demonstrations;
- tests equipment required for demonstrations and student activities. Students with recognized training or representatives from industry could assist in the set-up.

Prior Knowledge Required

- Hardware knowledge from Activity 1
- Internet search skills

Teaching/Learning Strategies

- The teacher discusses with students why an integrated circuit micro-controller requires an operating system. The need for an operating system is then reviewed for PLCs and PCs. The primary reason is to allow high level and low level programming input commands to perform desired functions.
- Present the overview of the progression of operating systems. Students highlight key terminology on the terms worksheet (Appendix 1.2.1 – Overview of Operating systems and Appendix 1.2.2 – Key Terms). The teacher explains the purpose of the operating system and provides students with a definition to write in their notebook.

-
- Teachers and students discuss the differences between high level and low level input into the different layers.
 - Students add a copy of the computer model to their notes (Appendix 1.2.3 – Computer Model).
 - Teachers demonstrate the boot-up procedure of the computer, explaining each sequential step (a pallet or projector is helpful). Finish the demonstration by describing how to find and launch application software.
 - The teacher reviews and demonstrates (if possible) a sample operating system used in a PLC or a programmable integrated circuit (PIC).
 - Teachers describe how the operating system includes both hardware and software (Appendix 1.2.4 – Operating System, Hardware and Software), e.g., detail the process of a print command.
 - Students then logon to a system and launch an application software program that allows them to write or draw their understanding of how an operating system interacts with hardware and software as the teacher circulates to provide feedback and direction. The completed document is filed in the student Portfolio.
 - Students complete an Internet search to research the future direction of operating systems (see Resources for web addresses). Students add the results of their research to their portfolio.

Assessment/Evaluation Techniques

- Learning Skills Checklist from Activity 1
- Check Portfolio for activities and complete activity checklist started in Activity 1.

Accommodations

In addition to accommodations mentioned in Activity 1, teachers:

- download various operating systems on desktop computers for student exploration (if resources allow);
- group students for enrichment to seek particular information on the Internet;
- provide handouts of any notes or overheads for clarification;
- provide trade magazines for article research.

Resources

Software

Operating systems (e.g., DOS, Windows 9x, Windows NT, System 6.x, 7.x, 8.x, Unix, or others)
 OESS software applications (e.g., Corel *WordPerfect*, *Microsoft Works*, *Appleworks*, etc..)

Print

Intel Corp. *The Journey Inside*. (training kit and video)

Norton, Peter. *Essential Concepts*. New York: McGraw-Hill Ryerson Limited, 1999.
 ISBN 0-02-804394-4

Operating System Manuals and reference texts

White, Ron. *How Computers Work*. Quebec, Canada: 1997. ISBN 01-56-276546-9

Web Sites

Microsoft – search site for Windows 95 tutorials
<http://www.microsoft.com>

Linux Operating System - comprehensive information and resources about Linux
<http://www.linux.org>

Zdnet On Line Magazine – Reviews, Links, Search Tool, Developer Section, Links to Mac Magazines and Other Magazines
<http://www.zdnet.com>

Infoworld Magazine – interviews, subject Index, Careers, Search Engine
<http://www.inforworld.com>

PC World Magazine - News, Reviews, How-To, Fileworld, Search Engine
<http://www.pcworld.com>

Appendix 1.2.1

Overview of Operating Systems

Operating System: A set of programs that manages the overall operation of the computer system. Particular functions include: Process Management, Input/Output, Memory Management, File Management, Networking.

- PC operating systems primarily began with DOS (Disk Operating System).
- A second layer of software is called BIOS (Basic Input Output System). In a PC this is an example of a programmable integrated circuit (PIC).
- When PCs became cheaper and more powerful, other operating systems that purported to handle memory management and multitasking more comfortably were O/S2 and Unix.
- Microsoft answered with Windows 3.1 and Windows for Workgroups which were really applications that ran on top of DOS; they used the GUI (Graphical User Interface), developed at Xerox and popularized by Apple.
- Windows 95, 98, NT (client and server versions) with increasing stability are very prominent.
- Unix is the favoured system of many computer scientists and environments with heavy multitasking, multi-user, and terminal-based systems.
- Linux is gaining popularity as it is a free Unix-type operating system created by Linus Tovalds with assistance of developers around the world.
- In industry many specialized computers called PLCs (programmable logic controllers) function using a PCs operating system or contain their own operating system.
- Teachers and students discuss school operating systems.

Appendix 1.2.2

Key Terms

Students are required to complete this table during class discussions

Key Terms	Description
Analog	
Application Software	
Byte	
Data	
Digital	
DVD	
Disk Drive	
Diskette	
Document	
Executing	
File	
Floppy Disk	
Hard Disk	
kilobyte	
megabyte	
Memory	
Operating System	
Program	
Software	
Software Layers	
Storage	
System Software	
User	

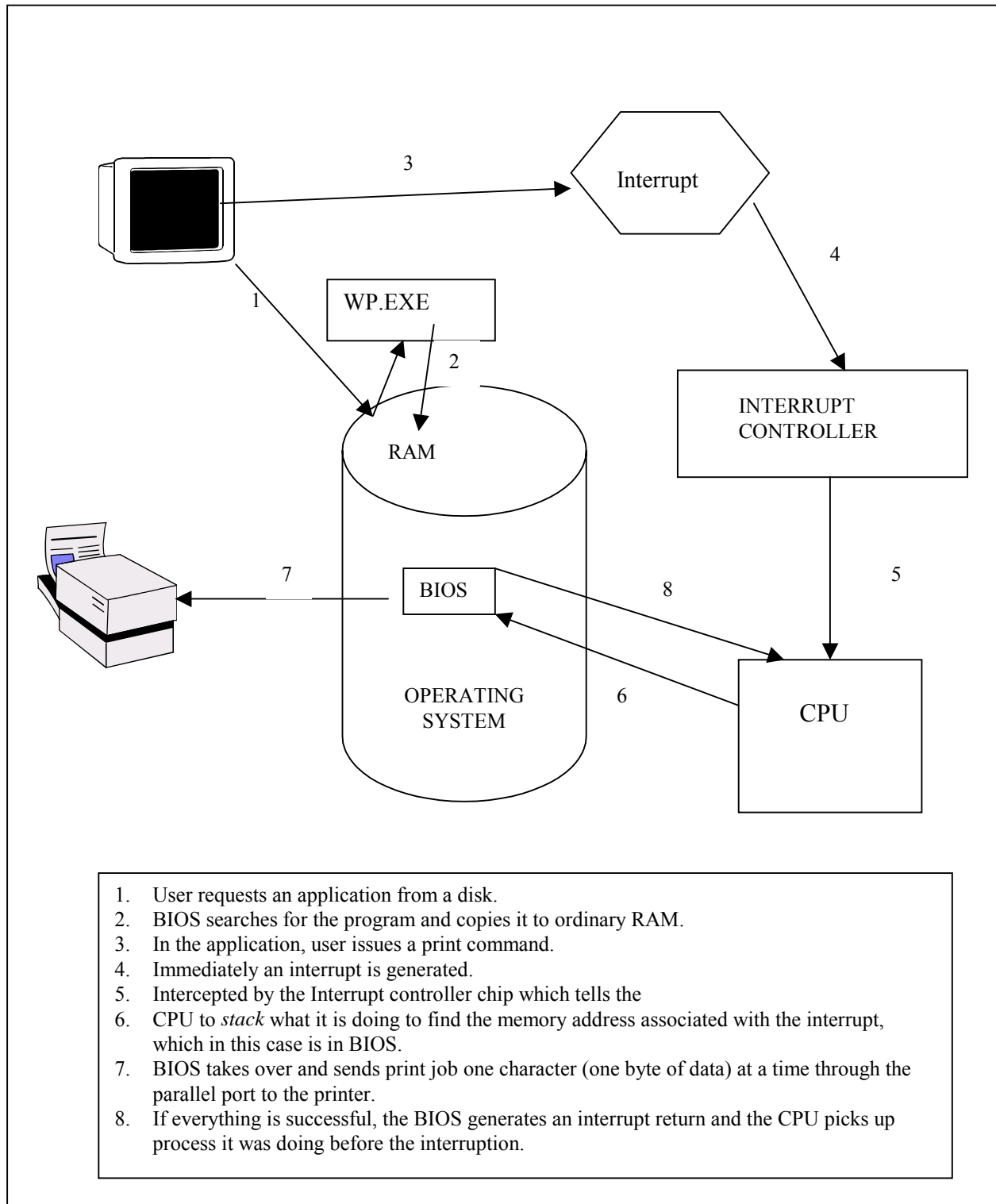
Appendix 1.2.3

Computer Model

```
*****
*           Application Software Layer           *
*           (Corel8, MSWord)                   *
*****
***
***
*** Binary Data passed between Application and OS layers
***
***
*****
*           Operating System Layer             *
*           ( MS-DOS, Windows 95)             *
*****
***
***
*** Binary data passed between OS and Hardware layers
***
***
*****
*           Hardware                           *
*           Components                         *
*****
```

Appendix 1.2.4

Operating Systems, Hardware and Software



Activity 3: Data Storage and Management

Time: 75 minutes

Description

Students are introduced to the concept of data (file) management within an operating system. Terminology such as drives, directories/sub-directories, files, folders, and volumes are used to illustrate effective strategies for managing various files (e.g., system files, programs, and data). Students set up their file management system (disk, server, or web site) with sub-directories to organize their notes, technical discussion, and glossary of terms. Students also learn data storage capacity and access speed as they apply to drives and storage systems.

Strand(s) and Expectations

Strand(s): Skills and Processes

Overall Expectations

SPV.02E - demonstrate the use of an operating system, including a network.

Specific Expectations

SP2.04E - use appropriate file management techniques.

Planning Notes

- This activity uses folder/sub-folder (or tree structure) file management concepts. File management is used throughout this profile and only basic concepts and tools are introduced.
- Any operating software may be used for this activity and still maintain the concepts and purpose of file management.
- Set up a student assignment in a read-only folder to distribute materials to students (if resources allow).
- Arrange to have disks available, if using them for students' files, or space on school server.

Prior Knowledge Required

- Computer engineering hardware from Activity 1
- Operating systems from Activity 2

Teaching/Learning Strategies

- Students complete the key terms form (Appendix 1.3.1 – Key Terms) as the teacher explains the purpose of file management and outlines a simple directory structure, file types, file sizes in bytes, and naming conventions.
- Students observe a demonstration on file management utilities (e.g., copy, rename, delete, move, create directory/folder) and label the structure diagram for their notes. (Appendix 1.3.2 – Sample Structure Label Descriptions). Students complete Exercises 1 and 2 (Appendix 1.3.3 – File Management Exercises) for homework. Some students with prior knowledge in this area may assist other students in completing the file management exercises.

-
- Teachers demonstrate how to access the computer system properties. Students access a PC's operation system data and record operating system, computer system, memory available, free hard disk space, CD-ROM drive, video card, sound card, peripherals, and network connection definitions (if networked). Some students may be familiar with this process from prior experience. As an enrichment activity, students may design a file management system for a school network on paper for homework.
 - Students and teacher discuss the size of storage systems (e.g., megabytes and gigabytes) and speed terms (e.g., nanoseconds) as they apply to various drives.

Assessment/Evaluation Techniques

- Periodically assess student's file management skills over the length of the course.
- Evaluate Exercise 2 (suggested mark out of 10).

Accommodations

- Create a poster or model on file naming conventions.
- Use grouping as a means of allowing stronger students to develop enhanced work and/or for peer teaching.
- Provide charts of speeds and storage capacities.
- Provide alternative metaphorical visuals for students (perhaps filing cabinet).

Resources

Norton, Peter. *Introduction to Computers*. New York: McGraw Hill. ISBN 0-02-804386-3

Parsons, Oja. *Computer Concepts*. ISBN 0-7600-3440-0

Appendix 1.3.1

Key Terms

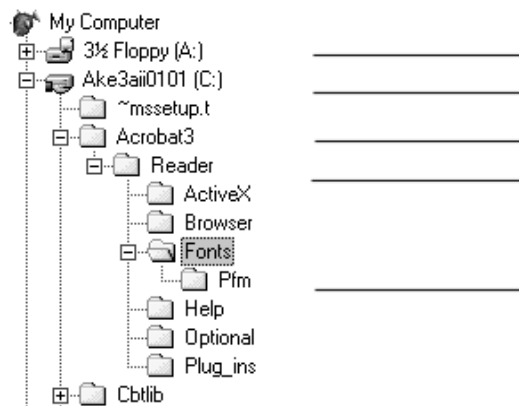
The list of key terms below are generic to most operating systems. More key terms must be added by the teacher, once an operating system is selected.

Key Terms	Description
Copy	
Delete	
Directory	
Directory Tree	
Drag and Drop	
File	
File Size	
Folder	
Hierarchical File Structure	
Icon	
Move	
Naming Convention	
Paste	
Rename	
Root Directory	
Sub Directory	
Other	

Appendix 1.3.2

Sample Structure Label Descriptions

Hand out for the lesson on file structures. Students fill in the information from the board.



Appendix 1.3.3

File Management Exercises

Exercise 1

Purpose:

Create a directory structure of your bedroom.

Procedure:

You need to organize your bedroom. On a sheet of paper, *draw out a directory structure* of your room. Refer to your notes on file name convention when *assigning file names* and directory names. Use a naming convention that people can *easily understand*.

Here are some *guidelines* to help you get started.

- 1) The room itself is the root directory.
- 2) List all the storage areas (shelves, drawers, desk, etc.) in the room. These are your sub-directories.
- 3) Add extra sub-directories. For example the book shelf can be divided into sub-directories (e.g., horror, comics, etc.).
- 4) Hand in for teacher sign-off.

Appendix 1.3.4

File Management Exercise 2

Purpose:

In this exercise you *create a directory structure* for your work during this course using either a disk or student workspace assigned to you. You also retrieve a file from a central folder to move to your workspace.

Procedure:

- 1) Using the method assigned by your teacher (disk or workspace), create a directory structure that includes Unit 1, Unit 2, Unit 3, Unit 4, and Unit 5.
- 2) Retrieve a file as directed by your teacher and move it to Unit 1.
- 3) Rename this file, including the date in the file name.
- 4) You may wish to create new directories (folders) for additional information (e.g., glossary, Internet web sites)
- 5) Have the teacher view your work for exercise sign-off.

Activity 4: Nomenclature

Time: 225 minutes

Description

Students inventory the hardware and software in the lab to create a database of equipment and resources and an individual system log for each PC (and PLC if available). Included in the inventory are operating system, computer system, memory, and if applicable, free Hard Disk Drive (HDD) space, CD (Compact Disk) drives, video card, sound card, peripherals, network type, etc. Students make decisions on software installation based on the system requirements. Finally, students learn about the functions of other computers that are most common in industry, namely, micro-controllers and programmable logic controllers.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes

Specific Expectations

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;
SP2.04E - use appropriate file management techniques.

Planning Notes

- Test the functionality of the operating systems.
- Prepare equipment for student use such as PLCs and PCs.
- Prepare a handout that includes a software title and the system requirements. An example is provided in Appendix 1.4.1, however, this depends largely on resources available.
- Obtain the Skills Inventory from the Conference Board of Canada or the Guidance or Co-operative Education Departments in the school.

Prior Knowledge Required

- Knowledge from Activities 1, 2, and 3
- Database application software skills

Teaching/Learning Strategies

- Teachers explain the need to keep clear concise records of computer information in order to facilitate repairs, add devices, upgrade application software and operating systems, and practise organizational skills required by people working in computer engineering careers.
- The teacher reviews how to handle machines properly. Complete safety lesson before students begin the inventory process (e.g., be sure all machines are unplugged, disconnected, grounded). Complete required activities on Safety Passports.
- Teachers and students determine what information should be kept in a lab database and what information should be included in a computer terminal log. Teachers facilitate discussion to keep it focussed and to ensure choices are valid by questioning.
- Divide the class in half: one group to inventory machines and create the class database and the second group to create the computer terminal logs required. Editing and proofreading are essential in this task.
- As students finish their part of the group task, they make decisions on whether or not the computers they have inventoried meet the system requirements for various software applications (e.g., Appendix 1.4.1 – Software System Requirements).

Assessment/Evaluation Techniques

- Learning skills checklist during class and group activity (Appendix 1.1.8)
- Evaluate the individual decision-making assignment (suggested 10 marks).
- Use student-teacher conferences to discuss initial skill inventory. (This may take several days and does not need to be complete by the end of this unit).
- Students complete a group work assessment sheet (Appendix 1.4.2 – Group Work Assessment).

Accommodations

- Provide a decision-making structure for students to follow.
- Create a poster or model of system requirements or obtain brochures from manufacturers.
- Students may inventory equipment for other departments or areas of the school as a school community service.
- Provide a word list.
- Encourage students to ask others they trust about their skill levels (other teachers, parents).
- Post safety rules and regulations.

Resources

Norton, Peter. *Introduction to Computers*. New York: Glencoe/McGraw Hill, 1994.
ISBN 0-02-804386-3

Appendix 1.4.1

System Software Requirements

Web-Page Designer		Our System Requirements Met	
		Yes or No	Require?
Operating System	Windows 95 or higher		
Computer System	Windows compatible 486 or higher PC		
Memory	8 MB		
Free Hard Disk Space	12 MB		
Peripherals	Microsoft compatible mouse		
	Modem		
CD-ROM	Required		
Decision and Reasons:			

Appendix 1.4.2

Group Work Assessment

What did you learn during this activity?	
What problems did you encounter?	
How did you solve these problems?	
What would have made this activity easier for you?	
If a rubric was supplied, assign yourself a level. If no rubric was supplied, assign yourself a grade out of 20 marks.	
Comments:	

Activity 5: Skills and Career Opportunities

Time: 120 minutes

Description

Students research and review careers in computer engineering locally and internationally and create a database of companies. Students also begin a personal skills inventory and assess their employability skill level in order to set personal goals for skill improvement.

Strand(s) and Expectations

Strand(s): Impact and Consequences

Specific Expectations

ICV.03E - identify related computer careers;

IC1.10E - identify computer skills that are important to employers.

Planning Notes

- Obtain the Employability Skills Training Brochure from the Conference Board of Canada or similar skill training information from the school Student Services centre.
- Prepare skill handouts.
- Have local telephone books and newspapers available.
- Obtain copies of national computer trade magazines and newspapers.
- Review Internet sites which post career opportunities in computer engineering.

Prior Knowledge Required

- Goal setting knowledge

Teaching/Learning Strategies

- Introduce students to the three skill areas that are critical to the Canadian workforce: Academic Skills, Personal Management Skills, and Teamwork Skills as outlined by the Conference Board of Canada. Discuss the skills that can be found under each board category. As a class create a skill rating system using a 1-4 scale. (An example is 1 = knowledge of this skill, 2 = can apply this skill sometimes, 3 = usually competent in this area, 4 = very confident in this area.)
- Students assess themselves on various skills under the three broad categories. They identify skills from each category that need improvement. Finally, they set personal goals in all three areas to be kept in their portfolio (Appendix 1.5.1 – Skills Inventory).
- Using local phone books, co-op resources, and the Internet, students research various local computer suppliers, service technicians, training centres, Internet providers, wholesalers, and manufacturers to establish the size of the local computer industry. Students create a personal database of company names, addresses, and phone numbers that are of particular interest to them to begin a resource list of potential contacts in the computer industry.
- Inventory students' research to create a pie chart of the computer industry and to assist students in comprehension of data organization and representation.

Assessment/Evaluation Techniques

- Use student-teacher conferences to discuss initial skill inventory. (This may take several days and does not need to be complete by the end of this activity) The chart is set up to allow students to check their skill level a second time at the end of the course.

Accommodations

- Encourage students to ask others they trust about their personal employability skill level (other teachers, parents/guardians, etc.).
- Provide an organizer for database of local companies.

Resources

Web Sites

Canada WorkinfoNet – source of career, education, and labour market information for Canadians
<http://www.workinfo.net.ca>

Canlearn Interactive – Human Resource Development Canada: career information
<http://www.canlearn.ca>,

Human Resource Development Canada – career information
<http://www.hrdc-drhc.gc.ca/maps/national/canada.shtml>

Information by National Occupation Classification Code
<http://www.eoa-hrhc.com/3519/menu/occnoc.stm>

Ontario Association of Certified Technicians and Technologists – education and careers
<http://oacett.org/>

Canadian Association of Professional Engineers – education and careers
<http://www.apegga.com/>

The Toronto Star Newspaper
<http://www.thestar.com/>

The Toronto Sun Newspaper
<http://www.canoe.ca/TorontoSun/home.html>

Other

Local telephone books and newspapers

Computer trade magazines and national newspapers

School Library/Resource Centre

Co-operative Education Department

Employability Skills Training Brochure, developed by the Conference Board of Canada

Appendix 1.5.1

Skills Inventory

Criteria for assessment: Level 1 = never, Level 2 = sometimes, Level 3 = mostly, Level 4 = always and with confidence

Academic Skills

Skill Description	Level				Skill Description	Level			
	1	2	3	4		1	2	3	4
Read and understand written material					Read and understand written material				
Write effectively					Write effectively				
Present own ideas to others					Present own ideas to others				
Understand oral information					Understand oral information				
Comprehend basic math					Comprehend basic math				
Organize data					Organize data				
Possess computer skills					Possess computer skills				
Use a calculator to solve problems					Use a calculator to solve problems				
Use tools and equipment					Use tools and equipment				
Read and understand technical materials					Read and understand technical materials				
Accept ideas of others					Accept ideas of others				
Research skills					Research skills				
Learn through observation					Learn through observation				
Make decisions					Make decisions				
Think independently					Think independently				
Create quality products					Create quality products				
Generate ideas					Generate ideas				
Learn quickly					Learn quickly				

Appendix 1.5.1 (Continued)

Skills Inventory

Personal Management Skills

Skill Description	Level				Skill Description	Level			
	1	2	3	4		1	2	3	4
Attend school daily and on time					Attend school daily and on time				
Meet school work deadlines					Meet school work deadlines				
Demonstrate self-control					Demonstrate self-control				
Show initiative					Show initiative				
Time-management skills					Time-management skills				
Aware of safety concerns					Aware of safety concerns				
Work well with others					Work well with others				
Self-esteem and confidence					Self-esteem and confidence				
Honest					Honest				
Integrity					Integrity				
Positive attitude					Positive attitude				
Persistence					Persistence				
Respect for others					Respect for others				
Creative ideas					Creative ideas				
Able to prioritize					Able to prioritize				
Upgrade skill levels					Upgrade skill levels				
Money management					Money management				
Sense of humour					Sense of humour				

Appendix 1.5.1 (Continued)

Skills Inventory

Teamwork Skills

Skill Description	Level				Skill Description	Level			
	1	2	3	4		1	2	3	4
Listen to others					Listen to others				
Identify non-verbal clues					Identify non-verbal clues				
Try new things					Try new things				
Dedicated					Dedicated				
Co-operative					Co-operative				
Plan with others					Plan with others				
Work toward group's goals					Work toward group's goals				
Exercise give and take					Exercise give and take				
Leadership					Leadership				
Encourage participation					Encourage participation				
Concern for others					Concern for others				
Responsible					Responsible				
Speaks to groups					Speak to groups				
Identify group problems					Identify group problems				
Negotiation					Negotiation				
Supportive of group					Supportive of group				
Brainstorming					Brainstorming				
Role model					Role model				

Unit 2: Integrated Circuits

Time: 20 hours

Unit Description

The focus of this unit is on integrated circuits and how the internal workings of a computer represent data such as characters and numbers. Students learn standard codes for internal numbering and character representation. They learn to design and construct fundamental logic gates (i.e., AND, OR, NOR, NAND, NOT, XOR, XNOR). They also learn about and construct simple electronic circuits, apply Boolean algebra, and devise truth tables to test and describe their functionality. Students develop an understanding of gates, semi-conductors (e.g., transistors, diodes, etc.), and integrated circuits by designing and building simple logic gates.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations: TFV.01E, TFV.02E, TFV.03E, TFV.04E; SPV.03E; ICV.01E; ICV.02E, ICV.03E.

Specific Expectations: TF1.01E, TF1.02E, TF1.03E; TF2.01E, TF2.02E, TF2.03E, TF4.04E, TF3.02E; SP1.01E, SP1.02E, SP1.03E; IC1.02E IC1.03E, IC1.04E.

Activity Titles (Time + Sequence)

Activity 1	Development of Computer Engineering Technology	120 minutes
Activity 2	Understanding Gates and Constructing Circuits	180 minutes
Activity 3	Number Systems	240 minutes
Activity 4	Wiring the Fundamental Gates	480 minutes
Activity 5	Relating Gates to Hardware and Software	180 minutes

Prior Knowledge Required

Students have some macro computer hardware experience from Unit 1. Most have some keyboarding background and experience with the use of common Ministry-licensed application software such as a word processor. All students have studied Grade 9 Science, The Characteristics of Electricity or Electrical Applications. Students must demonstrate knowledge of shop safety. Students may also have benefited from having taken Integrated Technologies in Grade 9, specifically the electronic poster activity.

Unit Planning Notes

The integrated circuits, conductors, and related electronic components necessary to complete the hands-on activities must be organized. Safety precautions for students must be covered for the wiring of the logic gates such as use of safety glasses, voltage considerations, and handling of small components. Use the Safety Passport created in Unit 1 to ensure all students understand necessary safety precautions.

Teaching/Learning Strategies

The unit is best implemented through hands-on activities that explore the relationship between integrated circuits and the binary number system. Limited hardware resources necessitate that the concepts be introduced theoretically. Other teaching/learning strategies include brainstorming, conferencing, class discussions, independent and group work, process notes, and Socratic lessons.

Assessment and Evaluation

Formative evaluations may help identify knowledge and skill levels and allow teachers to make sure that all students are challenged. A mixture of assessment strategies including practical assignments, quizzes, tests, and unit projects is used to gather assessment information towards a summative unit evaluation.

Resources

Software

Current Ministry of Education licensed software to record results of the hands-on activities

Electronic Workbench

Print

Baker, J. *Digital Computer Technology: An Introduction*. Toronto: McGraw-Hill Ryerson, 1983.

ISBN 0-13-211947-1

Blissmer, Robert H. *Introducing Computers*. New York: John Wiley & Sons, Inc., 1991.

ISBN 0471-53443-9

Lawrence, Orville. *Computer Technology*. Toronto: McGraw-Hill Ryerson, 1984.

ISBN 0-07-548711-X

Shelly, Gary and Thomas Cashman. *Computer Fundamentals for an Information Age*. California: Anaheim Publishing Co., 1984. ISBN 0-88236-125-2

Simms, Forrest. *Getting Started in Electronics*. USA: Radio Shack, 1983. Cat No 276-5003

Web Sites

Search Engine

<http://sideclick.switchboard.com>

Activity 1: Development of Computer Engineering Technology

Time: 120 minutes

Description

Students focus on hardware aspects of the various generations of computers presented in Unit 1. Students also review the social impact of ever-changing computer technology on individuals and society.

Strand(s) and Expectations

Strand(s): Impact and Consequences

Overall Expectations

ICV.01E - describe the evolution of computer electronics;

ICV.02E - identify the social impact of computers and associated technologies.

Specific Expectations

IC1.03E - identify important scientific advances in computer electronic components;

IC1.04E - describe the development of computer engineering technology and its impact.

Planning Notes

- Identify the electronic components of each computer generation and create a presentation of information (overheads, *PowerPoint*), beginning in 1971 with the first microprocessor chip up to and including recent changes. Research the function of semi-conductors and how they are used in a logic circuit. Be sure to include architecture and cabling improvements during this time frame (see Resources).
- Find a current news article on digital technology (e.g., Consumer Electronics Show).
- Organize research materials for social and environmental impact study.
- Decide due date for independent study on social and environmental consequences of changing computer engineering technology.

Prior Knowledge Required

- Keyboarding skills
- Use of some application software
- Hardware terminology from Unit 1

Teaching/Learning Strategies

- Begin this lesson with segment 5 of *The Journey Inside* video (see Course Overview Resources) (8 min 50 sec). This introduces the topic of computer chip technology. Students retrieve their notes on historical developments of computer systems from Unit 1 and add additional information during class discussion and teacher presentation.
- The teacher and students discuss various aspects and terms used in the development of computer technology including IC, digital signals, analog signals, transistors, etc.
- Introduce students to the news article on digital technology to spark a discussion on the importance of scientific advances in computer electronics. Students gain awareness from each other and have an opportunity to share information about products and technology they have encountered.
- Facilitate the discussion to include the social impact of some of the new technologies (e.g., digital cameras in watches, pendant radios). Introduce the independent study including depth, scope, and presentation style (Appendix 2.1.1 – Independent Study on Social and Environmental Consequences of Changing Computer Engineering Technology).
- Direct students to appropriate resources and assist with ideas for study.

Assessment/Evaluation Techniques

- Learning Skills Checklist from Unit 1
- Regularly conference with students on the progress of their independent study to monitor progress and offer suggestions for next steps.
- Summative assessment using the independent study rubric to evaluate student independent study

Accommodations

- Provide adaptive physical devices for identified students.
- Provide application software as needed.
- Assist students filtering information as necessary.
- Encourage students to share their knowledge and skills as peer mentors.
- Encourage alternative presentation methods, other than a written report, to allow for multiple intelligences.

Resources

Print

Shelly, Gary and Thomas Cashman. *Computer Fundamentals for an Information Age*. California: Anaheim Publishing Co., 1984. ISBN 0-88236-125-2

Web Sites

Global Network of Environment and Technology

<http://www.gnet.org>

Environment Canada's Adaptive Computer Technology program

http://www.dinf.org/csun_98_125.htm

The Toronto Star

<http://www.thestar.com>

Appendix 2.1.1

Independent Study on Social and Environmental Consequences of Changing Computer Engineering Technology

The intention of the independent study assignment in this unit is to allow students to explore in depth an area of changing computer engineering technology that is of particular interest to them. As the leaders of tomorrow, it is important for students to be able to understand the social and environmental impact of the exponential change in computer engineering technology. Information gained from this assignment impacts important choices in your life.

The Assignment

Research one change in computer engineering technology. Explore how it has impacted society and our environment (e.g., digital electronics, semi-conductors (i.e., transistors), increased data storage capability, faster access data storage, faster microprocessors, expanding applications for microprocessors, etc.). Include:

- your specific computer engineering technological change;
- how it evolved (need/want);
- when, where, and who created the change;
- research past effects of the development on society and the environment and possible projected impacts;
- your opinion, supported by your research, of the positive or negative impact;
- where you see this change in the future of computer engineering;
- bibliography of sources.

Presentation

You may present your conclusion through:

- written report;
- multimedia presentation;
- demonstration;
- video;
- animated depiction;
- other ideas approved by your teacher.

Appendix 2.1.1 (Continued)

Independent Study on Social and Environmental Consequences of Changing Computer Engineering Technology

Please refer to this rubric, as you work on your independent study, to assess your own work.

Independent Study Rubric

Criteria	Level 1 (50-59%)	Level 2 (60-69%)	Level 3 (70-79%)	Level 4 (80-100%)
Knowledge and Facts IC1.03E	- demonstrates limited knowledge of the chosen computer engineering technology change - limited knowledge of social or environmental impact	- demonstrates some knowledge of the chosen computer engineering technology change - some knowledge of social or environmental impact	- demonstrates considerable knowledge of the chosen computer engineering technology change - considerable knowledge of social or environmental impact	- demonstrates thorough knowledge of the chosen computer engineering technology change - thorough knowledge of social or environmental impact
Thinking/ Inquiry IC1.04E	- one piece of evidence to support position	- more than one piece of evidence to support position	- sufficient use of evidence to support position	- in-depth use of evidence to support position
Communication <ul style="list-style-type: none"> • Appropriate choice of medium for presentation • Bibliography 	- one presentation medium was used - bibliography somewhat formatted	- more than one presentation medium for topic - bibliography is mostly formatted	- sufficient use of presentation media to enhance presentation - bibliography format is correct and adequate	- presentation medium enhances topic and is memorable - bibliography is complete, in-depth, and reflects a variety of resources
Application IC1.04E ICV.02E	- limited awareness of societal issues	- adequate awareness of societal issues	- successful extension to societal issues	- expert extension to societal issues with mature insight

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Activity 2: Understanding and Constructing Gates and Circuits

Time: 180 minutes

Description

Students learn the fundamental components and concepts needed to safely wire fundamental electronics. The components and concepts include digital and analog signals, alternating current (AC), direct current (DC), voltage, breadboards, gates, semi-conductors, switches, integrated circuits, pin diagrams for individual pins circuits as well as related circuit diagrams and schematics.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations

TFV.01E - describe how the internal components of the computer enable the peripherals to function;

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.04E - describe and illustrate the functions of logic gates;

SPV.03E - use logic gates to construct simple circuits.

Specific Expectations

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;

TF2.02E - identify the basic internal and external components of a computer;

TF2.03E - describe the function of each basic component;

TF2.04E - identify computer internals and peripheral devices and describe their relationships;

SP1.02E - derive the truth tables of the fundamental logic gates;

IC1.02E - use safe practices in the handling of computer hardware and electronic components.

Planning Notes

- Inventory electronic components available.
- Review the precise terminology and application for each component and system.
- Prepare example circuit diagrams and schematic examples (Appendix 2.2.3 – Logic Gates Summary).
- Review safety features of equipment and classroom.

Prior Knowledge Required

- Grade 9 Science and Integrated Technologies (recommended)
- Students should have clear understanding of safe handling of computer hardware from Units 1 and 2.

Teaching/Learning Strategies

- Show segment 2 of *The Journey Inside* video (7 min 5 sec – see Course Overview Resources). Students gain a basic understanding of the four components of a circuit: the conducting path, electrical load, power source, and switch. Spend time showing students the various semi-conductor components that are mentioned, (i.e., transistor, resistor, light emitting diode (LED), capacitor). Students record a note on the function of each.
- Introduce students to And and Or statements using familiar scenarios (Appendix 2.2.1 – Introduction to Logic Gates, Circuits, Truth Tables).
- Teach a safety lesson on the safe handling of components, voltage, connection to electrical current, and use of tools required for soldering. Sign off Safety Passports.
- Teachers review basic soldering techniques.

-
- As a hands-on activity, build simple circuits (using single throw mechanical switches), asking the same questions as in the scenarios and building truth tables (Appendix 2.2.2 – Logic Gates Circuits Activity).
 - Hand out the summary of symbols and truth tables for all seven common logic gates (Appendix 2.2.3 – Logic Gates Summary). Review all the material with students, explaining the relationship of inputs to outputs.
 - Students complete the questions to apply their knowledge of logic gate theory for homework (Appendix 2.2.4 – Questions).

Assessment/Evaluation Techniques

- Assessment of homework activity provides students with written/oral feedback on areas that need improvement for success.
- Formative assessment in the form of roving conferences if the activity is hands-on (not a demonstration)

Accommodations

- Place large poster in the room of Logic Gate Theory.
- Use small circuits for initial questions, moving to larger circuits.
- Quiz students after And, Or, and Not gates have been learned.
- Allow students the opportunity to redo questions after feedback.
- Keep one of each physical circuit available for students to view.
- Students can design their own circuits and determine the possible output.
- As enrichment, students may use transistors and diodes to create a semi-conductor circuit that represents one or more logic circuits.

Resources

Print

Blissmer, Robert H. *Introducing Computers*. New York: John Wiley & Sons, Inc., 1991.
ISBN 0471-53443-9

Simms, Forrest. *Getting Started in Electronics*. USA: Radio Shack, 1983. Cat No 276-5003

Appendix 2.2.1

Introduction to Logic Gates, Circuits, Truth Tables

Ask students to think about the following situations. They must decide what conditions are necessary for the outcome. They must then determine if all conditions must be met or if the outcome can be reached by satisfying one of the conditions.

- Jane wants to study computer engineering at College. She must have both math and Physics.
- Mr. Cane is giving his students an assignment. To complete the task students must have a bibliography and a video or other audio presentation.
- The alarm in your car will sound if you touch the vehicle or use the key fob clicker.
- There is an election for school council coming up. You may vote for the senior class president if you are in Grade 11 or 12 and you have a student card.
- Your stereo is plugged into a socket that is controlled by a light switch. For this reason you can leave your stereo on or turn it off.

Represent answers in a table form on the board or overhead.

Appendix 2.2.2

Logic Gates Circuit Activity

This next assignment can be a hands-on activity or teacher demonstration.

Gates and Truth Tables

1. Wire a simple circuit with a light bulb and a switch.
If switch is open is the light on?
If switch is closed is the light on?

Construct a simple truth table.

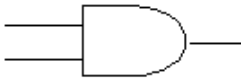
S1	Light
0 (open)	
1 (closed)	


2. Enhance circuit with two switches in series.
Repeat questions and create a truth table.
3. Repeat circuit with two switches in parallel.
Repeat questions and create a truth table.

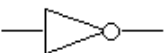
Introduce logic gate symbols and truth tables. See Appendix 2.2.3 for summary.


Appendix 2.2.3


Logic Gates Summary

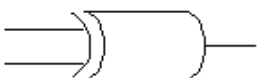
AND		A	B	$A \bullet B$
		0	0	0
		0	1	0
		1	0	0
		1	1	1

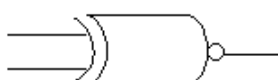
OR		A	B	$A + B$
		0	0	0
		0	1	1
		1	0	1
		1	1	1

NOT		A	\bar{A}
		0	1
		1	0

NAND		A	B	$\overline{(A \bullet B)}$
		0	0	1
		0	1	1
		1	0	1
		1	1	0

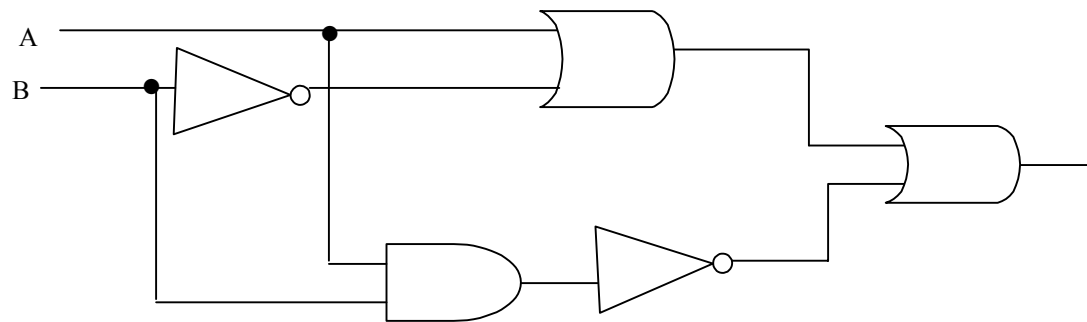
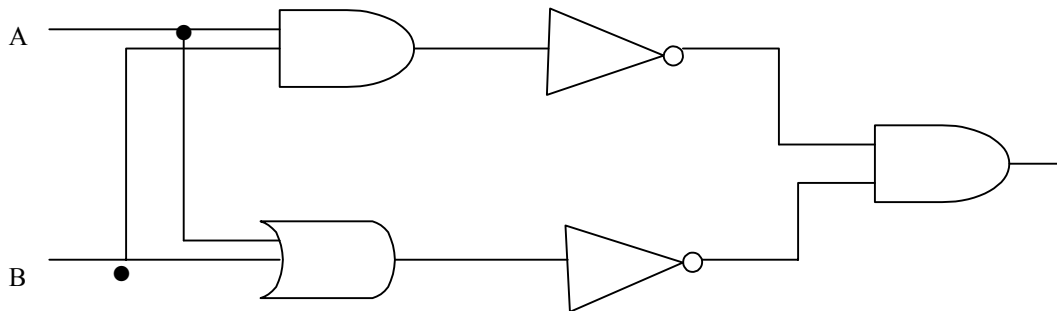
NOR		A	B	$\overline{(A + B)}$
		0	0	1
		0	1	0
		1	0	0
		1	1	0

XOR		A	B	$A \oplus B$
		0	0	0
		0	1	1
		1	0	1
		1	1	0

XNOR		A	B	$\overline{(A \oplus B)}$
		0	0	1
		0	1	0
		1	0	0
		1	1	1

Appendix 2.2.4

Logic Gates and Truth Tables



Activity 3: Number Systems

Time: 240 minutes

Description

Students develop the relationship between the binary number system and computer logic. Also, students learn how to convert Base 10 numbers into binary and vice versa. Character representation of binary codes is explored. Students have the opportunity to experiment in writing their own message and decoding.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Process

Overall Expectations

TFV.03E - explain internal numbering and character representation systems.

Specific Expectations

TF1.01E - describe the relationship between the binary number system and computer logic;

TF1.02E - define a standard way of representing characters in a binary code;

SP1.01E - convert positive integer numbers to binary form.

Planning Notes

- Review how programming software handles character representations.
- Have eight pennies for each pair of students and either a handout and/or overhead of bit information (Appendix 2.3.1 – Bit Information).
- Review binary and Base 10 conversions.
- Prepare coded messages for the students to decipher.
- Have copies of ASCII code available (both standard and extended).

Prior Knowledge Required

- Understanding of the decimal number system and place value

Teaching/Learning Strategies

- Check homework and clarify any questions in Activity 2.
- Show segment 3 of *The Journey Inside* video (8 min 25 sec – see Course Overview Resources). Students gain an understanding of how information is communicated through the use of codes.
- Hand each pair of students eight pennies and work through the questions on bit information (Appendix 2.3.1 – Bit Information). Ask students what pattern they can see forming in the right column (numbers double).
- Students are challenged to count as high as they can on one hand and told the answer is greater than ten. While students ponder the challenge, teachers demonstrate, with the aid of a simple series circuit, the binary logic states of ONE and ZERO (TRUE and FALSE, HIGH and LOW) by equating them to series circuit lamp ON and OFF condition.
- Binary numbers are introduced by initiating finger counting on one hand – no fingers up is zero, thumb up is a one, pointer finger up is two, index finger up is a four, next is eight, and little finger represents sixteen. Students demonstrate counting to thirty-one on one hand.
- This sets the stage for demonstrating how to convert numbers from Base 10 to Base 2 (binary) (Appendix 2.3.2 – Binary Number System Conversion). Work through several examples with students.

-
- Give students a quiz on binary conversion to assess their grasp of the concept (Appendix 2.3.2 – Binary Number System Conversion).
 - Handout the ASCII conversion information (Appendix 2.3.3 – ASCII Representation). Since computers cannot think like we do, they need a code to translate our language into data that it can process and then convert it back into recognizable language.
 - Students complete conversion exercises (Appendix 2.3.4 – Binary/Decimal Conversion Exercises).

Assessment/Evaluation Techniques

- Formative assessment of quiz at the end of the binary conversion exercise to prompt students on progress and show changes required for success of conversion application.
- Summative assessment of conversion exercises (Appendix 2.2.4 – Binary/Decimal Conversion Exercises).

Accommodations

- Use extensive visual aids and demonstrations to assist students as needed.
- Provide an enlarged copy of conversion methodology in classroom as well as ASCII character chart.
- Use a variety of teaching styles to accommodate learning styles.
- Provide appropriate adaptive devices or implementation accommodations for identified students.

Resources

Print

Baker, J. *Digital Computer Technology: An Introduction*. Toronto: McGraw-Hill Ryerson, 1983.
ISBN 0-13-211947-1

Blissmer, Robert H. *Introducing Computers*. New York: John Wiley & Sons, Inc., 1991.
ISBN 0471-53443-9

Lawrence, Orville. *Computer Technology*. Toronto: McGraw-Hill Ryerson, 1984.
ISBN 0-07-548711-X

Appendix 2.3.1

Bit Information

One coin	Heads	Tails	One bit can describe two choices
Two coins			
Three coins			
Four coins			
Eight Coins			

Any computer switch whether on or off is called a BIT (Binary digIT). It can have a value of either 1 or 0. Eight bits grouped together equals one character, a letter, number, or special symbol on the keyboard called a byte. A kilobyte is 1024 bytes, a megabyte is 1 048 576 bytes and a gigabyte is 1 073 741 824 bytes.

Appendix 2.3.2

Binary Number System Conversion

From Base 10 (decimal) To Base 2 (binary)

Method 1

Convert 156 decimal to binary.

$156/2=78$ $r = 0$ LSB
 $78/2=39$ $r = 0$
 $39/2=19$ $r = 1$
 $19/2=9$ $r = 1$
 $9/2=4$ $r = 1$
 $4/2=2$ $r = 0$
 $2/2=1$ $r = 0$
 $1/2=0$ $r = 1$ MSB

Read the remainders from bottom (most significant bit) to top (least significant bit)

156 decimal = 10011100

Note: r = remainder

LSB - Least significant bit

MSB - Most significant bit

Method 2

Convert 156 decimal to binary using the chart:

512	256	128	64	32	16	8	4	2	1
		1			1	1	1		

Choose the number that is closest to 156 but not over, put a one under it and subtract it from 156.

$156-128 = 28$

repeat above with new number 28, and so on until you reach 0.

$28 - 16 = 12$

$12 - 8 = 4$

$4 - 4 = 0$

Fill in remainder of chart with 0s. $156 = 10011100$ in binary.

From Binary To Base 10

Convert 1110010 to decimal

64	32	16	8	4	2	1
1	1	1	0	0	1	0

Add together all numbers from top row that have a 1 underneath

$= 64 + 32 + 16 + 2$

$= 114$

Quiz

Convert Binary to Decimal

10010 = _____

110110 = _____

101 = _____

Convert Decimal to Binary

231 = _____

63 = _____

2564 = _____

Appendix 2.3.3

ASCII Representation

ASCII was established to achieve compatibility between various types of data processing equipment. ASCII, pronounced “ask-kee”, is the common code for microcomputer equipment. The standard ASCII character set consists of 128 decimal numbers ranging from zero through 127 assigned to letters, numbers, punctuation marks, and the most common special characters. The Extended ASCII Character Set also consists of 128 decimal numbers and ranges from 128 through 255 representing additional special, mathematical, graphic, and foreign characters.

American Standard Code for Information Interchange (ASCII)

ROW BITS 765

0	1	2	3	4	5	6	7
000	001	010	011	100	101	110	111

COLUMN BITS 4321

0000

0001

0011

0100

0101

0110

0111

1000

1001

1010

1011

1100

1101

1110

1111

COMBINE BITS 765 with BITS 4321 gives ASCII code 110 0001 which is a lower case a.

Appendix 2.3.3 (Continued)

ASCII Representation

Control Characters 0 -31

NUL	Null	DLE	Data link escape
SOH	Start of Heading	DC1	Device control 1
STX	Start of text	DC2	Device control 2
ETX	End of text	DC3	Device control 3
EOT	End of transmission	DC4	Device control 4
ENQ	Enquiry	NAK	Negative acknowledge
ACK	Acknowledge	SYN	Synchronous idle
BEL	Bell (audible signal)	ETB	End of Transmission Block
BS	Backspace	CAN	Cancel
HT	Horizontal tabulation	EM	End of Medium
LF	Line feed	SUB	Substitute
VT	Vertical tabulation	ESC	Escape
FF	Form feed	FS	File separator
CR	Carriage return	GS	Group separator
SO	Shift out	RS	Record separator
SI	Shift in	US	Unit separator

Appendix 2.3.4

Binary/Decimal Conversion Exercises

Convert the following decimal (Base 10) numbers into binary (Base 2). Show all calculations.

126
54
89
1034
455

Convert the following binary (Base 2) numbers into decimal (Base 10). Show all calculations.

1000110
1110001
1000011
1100001
1011110

Using the ASCII character set, write the decimal equivalent and character for the following.

01010011
01110101
01110000
01100101
01110010
00100001

Write your own message and decode.

Activity 4: Wiring the Fundamental Gates

Time: 480 minutes

Description

Through hands-on activities, students construct circuits which represent the seven fundamental gates. By applying an input, processing, output problem-solving model to these activities, students develop truth tables and Boolean equations for each gate. Students have an opportunity to practise Boolean algebra.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.04E - describe and illustrate the functions of logic gates;

SPV.03E - use logic gates to construct simple circuits.

Specific Expectations

TF1.01E - describe the relationship between the binary number system and computer logic;

TF1.03E - describe the function of the fundamental logic gates, including the function of each pin: AND, NAND, OR, NOR, XOR, XNOR, and NOT;

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;

SP1.02E - derive the truth tables of the fundamental logic gates;

SP1.03E - write Boolean equations for the fundamental logic gates;

IC1.02E - use safe practices in the handling of computer hardware and electronic components.

Planning Notes

- Consideration must be given to the opportunities and limitations of available hardware and software.
- Use a copy of Appendix 2.2.3 and cut information into sections for use as a matching exercise at the beginning of the class.
- Gather electronic components.
- Inventory the integrated circuits containing the appropriate gates.
- Prepare step-by-step instructions if required.
- Safety considerations for the room and equipment should be noted
- Group work may be considered to complete the hands-on activities.
- Organize hardware to ensure security of activities.
- Consider electronic workbench to support circuit design.
- Select appropriate tools and processes to fabricate logic circuit (Appendix 2.4.3 – Block Circuit Diagram).

Prior Knowledge Required

- Grade 9 Science, Electricity Unit
- Comprehensive understanding of Activities 1, 2, and 3

Teaching/Learning Strategies

- Introduce students to the idea that logic gates can also be expressed in mathematical terms using Boolean algebra. Use the table (Appendix 2.4.1 – Boolean Algebra) to summarize the eight basic rules of Boolean algebra. Since this is a difficult leap for some students, remind them that they must know the equations for the seven fundamental gates only.

-
- Review safety considerations required before students begin wiring the gates.
 - Also, teachers review the role of transistors as semi-conductor switching devices used in logic circuits on printed circuit boards.
 - Review all equipment being used with students before the activity begins. Teacher selects the type of circuit which is constructed (e.g., breadboard, hard wire, or acid etched).
 - Use the cut up sections of the logic gates and truth table exercise for students to find their match. Once students have a match they wire the circuit with their partner. Each student must then write the truth table, draw the logic gate symbol, and write the Boolean expression for their circuit.
 - Ask each pair of students to choose three more logic gate circuits to wire and document. Reiterate that the documentation is done as an individual portion of the activity.
 - As a final exercise, ask students to design a logic gate circuit, which has three inputs, two outputs, and is a combination of at least four logic gates, and document the truth table (Appendix 2.4.3 – Block Circuit Diagram). This must be initialled by the teacher as complete. Next students wire their circuit design to prove their truth table. Students troubleshoot any inconsistencies that occur. The teacher circulates around the room to assist students who are having difficulties. This final activity must be handed in with appropriate documentation from each student. The final product could also be used as a student “show and tell” presentation.

Assessment/Evaluation Techniques

- Formative evaluation on homework and students’ notes from Activity 3
- Summative evaluation rubric (Appendix 2.4.2 – Wiring of Logic Gates Rubric)

Accommodations

- Use a variety of teaching styles to accommodate learning styles.
- Provide appropriate adaptive devices or implementation accommodations for identified students.
- Provide handouts to reinforce demonstrations.
- Provide supplementary print and/or audio-visual aids to support activities.
- Stronger students can build the circuits they designed in the previous activity.
- Use *Electronic Workbench* software for those having trouble with the hands-on component.

Resources

Software

Current Ministry of Education licensed software

Electronic Workbench.

Print

Baker, J. *Digital Computer Technology: An Introduction*. Toronto: McGraw-Hill Ryerson, 1983. ISBN 0-13-211947-1

Lawrence, Orville. *Computer Technology*. Toronto: McGraw-Hill Ryerson, 1984. ISBN 0-07-548711-X

Simms, Forrest. *Getting Started in Electronics*. USA: Radio Shack, 1983. Cat No 276-5003

Appendix 2.4.1

Boolean Algebra

Discovering the Rules of Boolean Algebra

Draw Logic Gate Symbol	Boolean Algebra Questions
OR (inputs A and 1)	If one of the inputs is always 1, what is the output? $A + 1 =$
OR (inputs A and 0)	If one of the inputs is always 0, what is the output? $A + 0 =$
OR (inputs A)	If $A = 0$ then the output is If $A = 1$ then the output is $A + A =$
OR + NOT at one input (inputs A)	If $A = 1$ then the output is If $A = 0$ then the output is $\bar{A} + A =$
AND (inputs A and 1)	If one of the inputs is always 1, what is the output? $A \square 1 =$
AND (inputs A and 0)	If one of the inputs is always 0, what is the output? $A \square 0 =$
AND (inputs A)	If $A = 0$, then the output is If $A = 1$ then the output is $A \square A =$
AND + NOT at one input (inputs A)	If $A = 1$ then the output is If $A = 0$ then the output is $\bar{A} \bullet A =$

Appendix 2.4.2

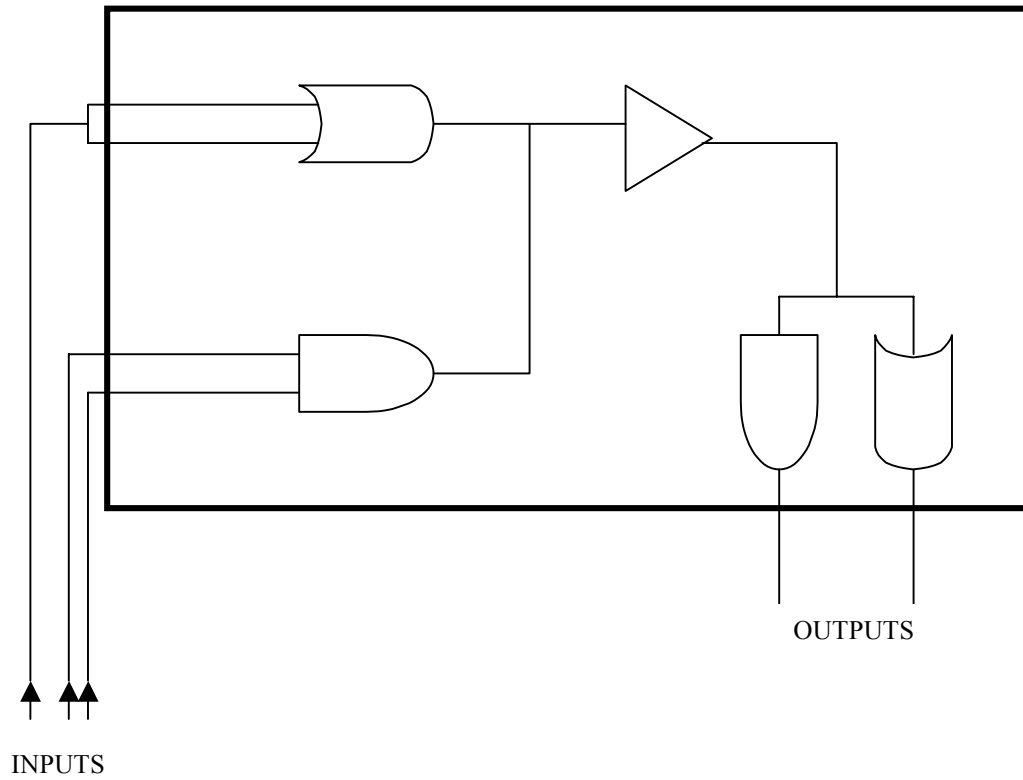
Wiring of Logic Gates Rubric

Criteria	Level 1 (50-60%)	Level 2 (60-70%)	Level 3 (70-80%)	Level 4 (80-100%)
Knowledge/ Understanding Describes and understands logic gates	- demonstrates limited ability in describing and illustrating the function of logic gates	- demonstrates some ability in describing and illustrating the function of logic gates	- demonstrates considerable ability in describing and illustrating the function of logic gates	- demonstrates thorough ability in describing and illustrating the function of logic gates
Describes the function of each pin of the logic gate	- demonstrates limited ability in describing and illustrating the function of each pin of the logic gates	- demonstrates some ability in describing and illustrating the function of each pin of the logic gates	- demonstrates considerable ability in describing and illustrating the function of each pin of the logic gates	- demonstrates thorough ability in describing and illustrating the function of each pin of the logic gates
Thinking/ Inquiry Derives Truth Tables for Logic Gates	- demonstrates limited ability deriving truth tables for logic gates	- demonstrates some ability deriving truth tables for logic gates	- demonstrates considerable ability deriving truth tables for logic gates	- demonstrates thorough ability deriving truth tables for logic gates
Writes Boolean Equations for Logic Gates	- demonstrates limited ability writing Boolean equations for logic gates	- demonstrates some ability writing Boolean equations for logic gates	- demonstrates considerable ability writing Boolean equations for logic gates	- demonstrates thorough ability writing Boolean equations for logic gates
Communication Teamwork skills	- demonstrates limited effectiveness as a team member	- demonstrates some effectiveness as a team member	- demonstrates considerable effectiveness as a team member	- demonstrates a high effectiveness as a team member
Problem solver	- demonstrates limited ability describing a problem-solving model	- demonstrates some ability describing a problem-solving model	- demonstrates considerable ability describing a problem-solving model	- demonstrates thorough ability describing a problem-solving model
Application Ability to wire logic gates	- demonstrates limited ability when wiring logic gates to construct circuits	- demonstrates some ability when wiring logic gates to construct circuits	- demonstrates considerable ability when wiring logic gates to construct circuits	- demonstrates a high ability when wiring logic gates to construct circuits
Observes safety procedures	- demonstrates limited ability in safety procedures	- demonstrates some ability in safety procedures	- demonstrates considerable ability in safety procedures	- demonstrates a high ability in safety procedures

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Appendix 2.4.3

Block Circuit Diagram



Activity 5: Relating Gates to Hardware and Software

Time: 180 minutes

Description

Students interrelate gates, circuit diagrams, truth tables, and Boolean equations of the seven fundamental gates. The micro investigation of hardware at the gate level is placed in context with the macro hardware components of Units 1 and 2.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.04E - describe and illustrate the functions of logic gates;

ICV.01E - describe the evolution of computer electronics.

Specific Expectations

TF1.01E - describe the relationship between the binary number system and computer logic;

TF1.03E - describe the function of the fundamental logic gates, including the function of each pin: AND, NAND, OR, NOR, XOR, XNOR, and NOT;

TF3.02E - describe how computers store and work with different types of data, including numbers and characters;

SP1.02E - derive the truth tables of the fundamental logic gates;

SP1.03E - write Boolean equations for the fundamental logic gates;

IC1.03E - identify important scientific advances in computer electronic components.

Planning Notes

- Organize multiple gate circuit diagrams.
- Research micro and macro computer components.

Prior Knowledge Required

- Understanding of fundamental circuit diagrams from Activity 4
- Relationships between these diagrams and truth tables and Boolean equations
- Understanding of hardware from Units 1 and 2

Teaching/Learning Strategies

- Teachers review with students a problem-solving model suited to computer engineering applications. Advise students that this activity uses the same rubric as Activity 4.
- Explain to students that this activity is the basis of all digital electronic circuits such as those used in computers and calculators. Everything to a computer is a number – either 1 or 0. The computer assembles all its switches to recognize this data and send it on to the central processing unit (CPU), which is an integrated circuit and microprocessor.
- Teachers explain to students that IC fabrication is a complex process that typically involves hundreds of processing steps using multiple chemicals, gases, temperatures, and machines. The latest ICs incorporate tens of millions of transistors and typically use hundreds of processing steps. The activity of constructing simple logic gate circuits helps students to comprehend how complex modern microprocessors are.

-
- Teachers introduce the functions of the output pins on a 25 pin parallel port, 9 pin output, and a serial port output (Appendix 2.5.2 – Pin Diagrams). This allows students to see how the PC sends and receives data when interacting with peripherals.
 - Before beginning the activity, re-emphasize any safety precautions for students (already signed on their passports).
 - Individually, students design a series of circuits to produce a simple mathematical formula such as $1 + 3 = 4$ using logic gate theory (Appendix 2.5.1 – Example of Simple Formula).
 - Students complete truth tables for each circuit. This is the basis of all calculators and computers.
 - Once students have their design on paper, they wire their circuit. Students may choose to create their design using *Electronic Workbench*.
 - Circulate to assist students in completing their task. Students hand in their written work for evaluation.

Assessment/Evaluation Techniques

- Formative assessment in the form of oral feedback as students are working on this activity
- Evaluate finished product and documentation using the rubric from Activity 4 (Appendix 2.4.2 – Wiring Logic Gates Rubric).
- Formative assessment using roving conferences to assist student groups and give them progress status and suggestions for success

Accommodations

- Use extensive visual aids and demonstrations to assist students as needed.
- Provide handouts to reinforce demonstrations.
- Provide supplementary print and/or audio-visual aids to support activities.
- Have students research how logic gates changed computers.
- Students may include Boolean algebra with circuit designs.

Resources

Software

Current Ministry of Education licensed software

Electronic Workbench.

Print

Baker, J. *Digital Computer Technology: An Introduction*. Toronto: McGraw-Hill Ryerson, 1983.
ISBN 0-13-211947-1

Lawrence, Orville. *Computer Technology*. Toronto: McGraw-Hill Ryerson, 1984.
ISBN 0-07-548711-X

Simms, Forrest. *Getting Started in Electronics*. USA: Radio Shack, 1983. Cat No 276-5003

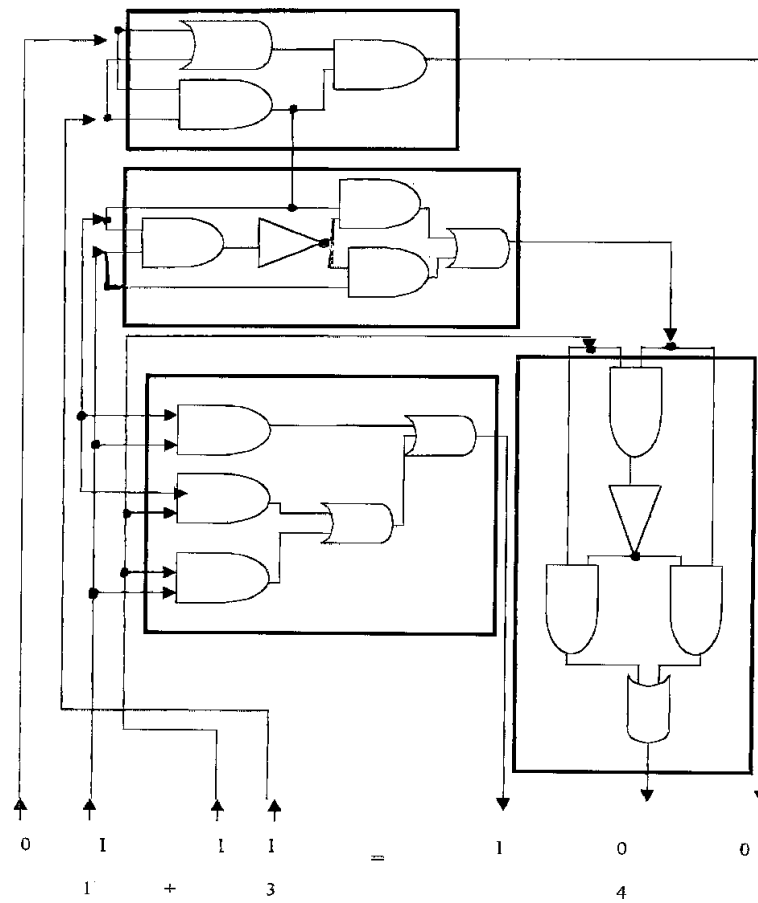
Web Sites

Search Engine

<http://sideclick.switchboard.com>

Appendix 2.5.1

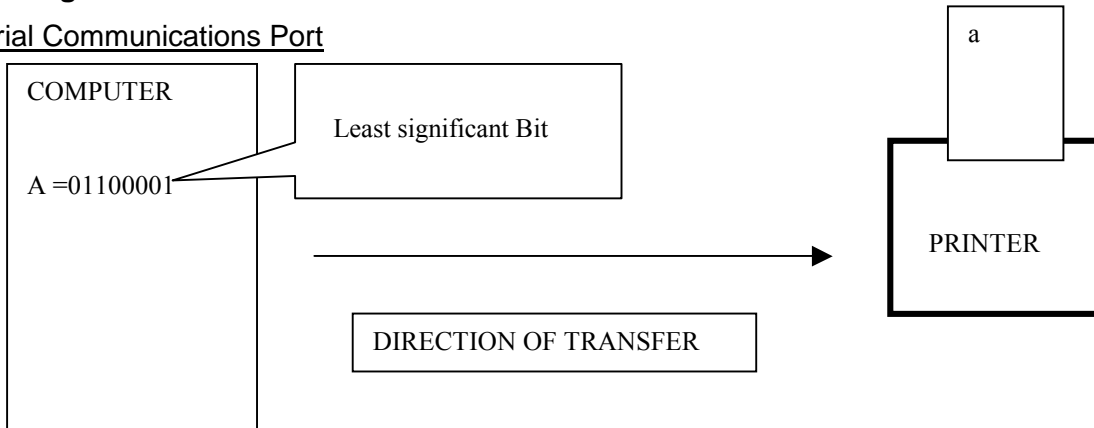
Example of Simple Formula



Appendix 2.5.2

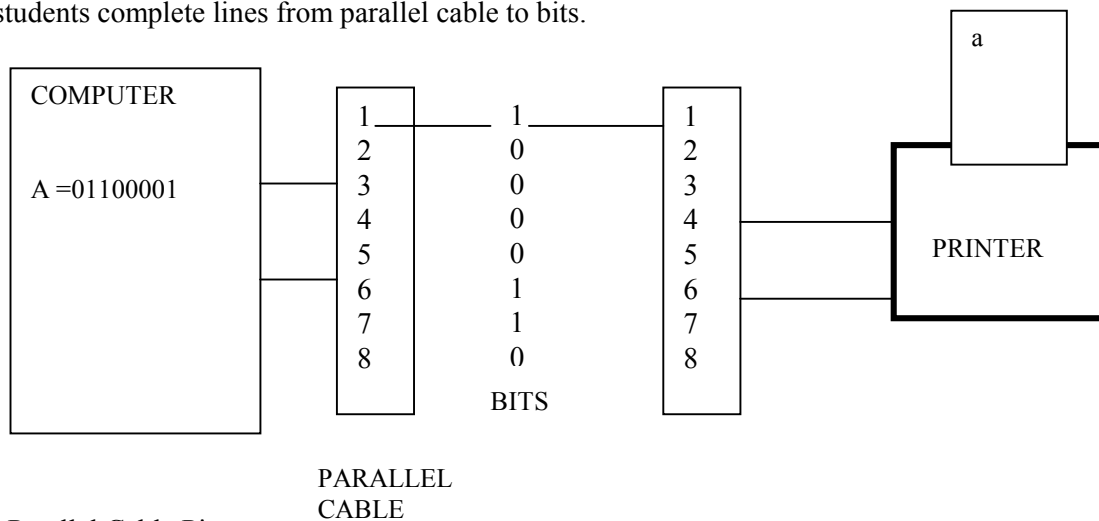
Pin Diagrams

Serial Communications Port



Parallel Communications Port

Have students complete lines from parallel cable to bits.



25 Pin Parallel Cable Pinout

- Pin 1 – strobe signal
- Pin 2-9 – data lines
- Pin 10 – acknowledgement of data received
- Pin 11 – Printer too busy
- Pin 12 – out of paper
- Pin 13 – select line
- Pin 14 – paper advance
- Pin 15 – error condition line
- Pin 16 – printer reset
- Pin 17 – On/off switch
- Pin 18-25 – ground wires

Unit 3: Networking

Time: 20 hours

Unit Description

Students explore and set up parallel and series computer communication processes within a computer and between computer systems (e.g., internal architecture, cabling standards, topology, and network types). They use problem-solving skills to apply their knowledge to tasks such as researching simple network types and building simple communication networks. Students also learn about the importance of network connectivity and infrastructure and how it impacts on our world as well as potential career opportunities in the area of computer networking.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations: SPV.01E, SPV.02E, ICV.01E, ICV.02E, ICV.03E, TFV.01E.

Specific Expectations: TF2.01E, TF2.02E, TF2.03E, IC1.02E, IC1.03E, IC1.05E, IC1.07E, IC1.08E, IC1.09E, IC1.10E, SP2.01E, SP2.05E, SP2.06E.

Activity Titles (Time + Sequence)

Activity 1	Setting up a Computer	150 minutes
Activity 2	Introduction to Networking Technologies	375 minutes
Activity 3	Activating a Computer on a Network	150 minutes
Activity 4	Network Options and Security	300 minutes
Activity 5	Networking	225 minutes

Prior Knowledge Required

Students have:

- basic computer hardware and operating system knowledge from Unit 1;
- word processing.

Unit Planning Notes

Students begin with exploratory hands-on activities then focus on underlying theory. Students complete integrated activities that unite both theory and practice. Ensure that computer resources can be dedicated for this unit. Ensure basic electronic tools and components are on hand for the construction activities. This is an opportunity to contact the school computer system operator (sysop) and/or the board computer technician to come in as a guest speaker. This unit may also be adapted to set up a mini-network, MIDI lab (Musical Instrument Digital Interface), or communications lab depending on school equipment. Contact local businesses to offer support through donation of materials, either hardware or written, or guest speakers outlining careers in networking. Learning stations addressing various concepts and skills may be used to offset difficult to obtain resources.

Teaching/Learning Strategies

- This unit incorporates demonstrations and hands-on activities.
- Other teaching and learning strategies include class discussion, brainstorming, Socratic lessons, peer and teacher conferencing, research, and activity instruction guides.
- If there is limited hardware available demonstrations, videos, experienced student assistants, industry professionals, or field trips should be used in conjunction with the activities.
- Students also explore avenues to careers.

Assessment and Evaluation

Within this unit some students may possess considerable prior knowledge and skills due to the growth of the Internet and the increasing popularity of home networks. A hands-up acknowledgement, short diagnostic quiz, or computer questionnaire such as the one used in Unit 1 assists in identifying the varied skill levels in order to make the curriculum challenging for all students. A mixture of assessment strategies (both formative and summative) should be used including assignments, skill demonstration, quizzes/tests, roving conferences, and presentations. The activity checklist and the Learning Skills Checklist from Unit 1 may be applied in this unit.

Resources

Software

Operating systems (e.g., DOS, Windows 9x, Windows NT, System 6.x, 7.x, 8.x, Unix, or others)

Ontario Educational Software Services [OESS software tools] (e.g., Corel *WordPerfect*, *Microsoft Works*, *Appleworks*, etc.)

Web and FTP server and client applications

Print

Feldman, Jonathon. *Sams Teach Yourself Network Troubleshooting in 24 Hours*. Sams, 1998.

ISBN 0672314886

Gregg, Kenneth. *Windows Networking Basics*. Harper Collins Canada, 1998. ISBN 0764532146

Kearns, Dave. *Sams Teach Yourself Windows Networking in 24 Hours*. Sams, 1998.

ISBN 0672314754

Keogh, Jim. *Core MCSE: Networking Essentials*. Prentice Hall of Canada Ltd.

ISBN 0130107336

Magendanz, Thomas and Radu Popescu-Zeletin. *Intelligent Networks: Basic Technology, Standards & Evolution*. International Thomson Press, 1996. ISBN 1850322937

MCSE Networking Essentials For Dummies, Training Kit. IDG Books Worldwide, 1999.

ISBN 0764506218

Network+ Certification Study Guide. Syngress Media, Inc., 1999. ISBN 0-07-211846-6

Operating System Manuals and reference texts

Video Resources

The Journey Inside

Intel Corporation – Part of *The Journey Inside Education* kit

Web Sites

Novell Network Primer – networking primer with links and a glossary

<http://www.novell.com/catalog/primer/primer.html>

IT Careers – careers search

<http://www.itcareers.com>

Intel Resources – teacher and student resources

<http://www.intel.com/education/k12/resources/index.htm>

Cisco Certification CCIE – Cisco certification requirements

<http://www.cisco.com/warp/public/625/ccie/>

Microsoft Educational Resources

<http://www.microsoft.com/education/instruction/default.asp>

3Com's Netprep program – school to work curriculum opportunities

<http://education.3com.com/Netprep/index.html>

Globe and Mail: Canada's National Career Site – career search

www.globecareers.com

Canada Employment Search Network – career search

www.canjobs.com

Canadian Career Page – career search

www.canadiancareers.com

Protecting Your PC and Media Against Computer Viruses

www.utoronto.ca/security/virus.htm

Computer Viruses – virus protection and detection

www.ccs.queensu.ca/pubs/itsnote/virus.html

Government of Canada, The Privacy Act – Intellectual Property Legislation

<http://www.privcom.gc.ca/privacyact1/htm>

Data Security – security on the Web

<http://www.cown.edu.au/ITDivision/security/password.htm>

Activity 1: Setting up a Computer

Time: 150 minutes

Description

Students set up a computer including installing available software. Students connect, configure, and test all peripherals. Finally, students troubleshoot any problems that arise. All students set up a PC; if resources are available a PLC may be configured and tested as well. The setup and installation of a micro-controller are also discussed.

Strand(s) and Expectations

Strand(s): Skills and Processes, Impact and Consequences

Overall Expectations

SPV.01E - connect and use correctly a variety of computer components and peripherals.

Specific Expectations

SP2.01E - set up a desktop computer system and install software;

IC1.02E - use safe practices in the handling of computer hardware and electronic components.

Planning Notes

- Prepare available samples of micro-controllers, PLCs, and PCs of various types.
- This activity focusses on setting up a desktop by having students install an operating system. Teachers can extend the concept of “setting up a computer” to setting up a micro-controller and PLC for a specific application such as a robot.
- Determine the most effective use of existing hardware within the recommended time allotment (e.g., two to three students per computer).
- Open an older discarded hard drive for demonstration purposes.
- If resources are limited, a single system may be set up several times to accommodate all students.
- This activity is done with stand-alone machines to not interrupt a networked environment.
- The teacher should review the procedures in the attached appendices. This activity assumes that the computer system hard drive has been configured (in CMOS set-up) prior to the installation of the operating system.
- The actual system installation can be performed as a class “walk through”. The teacher can modify the process to have the individual groups perform the set-up task.
- The teacher should review the disk partitioning, formatting, scandisk operations, and information available in the Help files of the operating system (see Resources).
- Inventory the operating system CD-ROM and software key.
- Ensure all software is available for the full installation including operating systems, device drivers, and application software.

Prior Knowledge Required

- Components of a computer system
- Correct terminology

Teaching/Learning Strategies

- Teachers and students review safety with static electricity and the importance of keeping contacts clean as they apply to components. Use the student Safety Passports (Appendix 1.1.2 – Sample Safety Passports).
- Begin the lesson with a review of the types of computers, e.g., micro processor, PLC, desktop and network workstation (Appendix 3.1.3).
- Review the safety considerations when setting up a desktop computer (grounded plugs, using power bars, dangling cords, eliminating the danger of static electricity, and unplugging power supply before opening a PC, etc.).
- The teacher explains how hard drives work in order to have some comprehension of the utility functions that students are required to complete by the end of the activity (Appendix 3.1.1 – Hard Drives).
- Students use the equipment they require to complete the task, including the monitor, CPU, keyboard, mouse, and a printer, if available. The teacher explains any special considerations they need to know (e.g., positioning of computers for plugs in the room). Students use this information to create a checklist for the activity.
- If available, students could set up a PLC to experience the setup requirements.
- Depending on the resources available, divide students into the appropriate number of groups. Students connect all the parts of their computer system. Circulate to help with troubleshooting and use questioning techniques to assist with problem solving.

-
- Once all components are connected, students load the operating system software (Appendix 3.1.2). If groups must wait to share the software resource, they should prepare the computer terminal log for their machine at this time, indicating what has been done. (Computer terminal logs were created in Unit 1). Students complete their personal checklist to keep in their portfolios.
 - All groups must then test their software to ensure their system is working and that all peripherals connected are functioning properly. Students fill in the group work assessment sheet (Appendix 1.4.2 – Group Work Assessment).

Assessment/Evaluation Techniques

- A formative assessment through student discussion and observation, encouraging students to assess their thinking for successful completion of task
- Assess student-created checklists, group work assessment sheets (Appendix 1.4.2 – Group Work Assessment), and terminal logs created in Unit 1. Provide students with written/oral feedback, to assist their success in upcoming related activities.

Accommodations

- Provide step-by-step instructions.
- Provide a glossary of terms.
- Provide visuals of different computer types.

Resources

Windows 95 Resource Kit, on the Windows 95 CD-ROM located in directory (/admin/reskit/helpfile/win95rk)

Blissmer, Robert H. *Introducing Computers*. New York: John Wiley & Sons, Inc., 1991.
ISBN 0471-53443-9

Norton, Peter. *Essential Concepts*. McGraw-Hill Ryerson Limited, 1999.
ISBN 0-02-804394-4

Appendix 3.1.1

Hard Drives

- The hard drive is one of the few components that is mechanical as well as electronic. It is contained in a sealed metal housing for protection from dust.
- The bottom of the drive contains a printed circuit board (known as logic board) that has various functions:
 - receives commands from the drive's controller (which is controlled by the O/S);
 - translates those commands to move the read/write heads across the platters;
 - makes sure the spindle turning the platters moves at a constant speed;
 - tells drive heads when to read and when to write to the disk.
- The number of platters and the composition of material coating them determine the capacity of the drive (e.g., 4.3 GB)
- When instructing the O/S to read or write a file, the hard disk controller moves to the drive's file allocation table (FAT). The O/S reads this information to determine where to start reading a file or where there is room to hold a new file.
 - A file could be in several clusters on several platters, but the FAT records a list of all the file's clusters.

Data Organization

- The disk contains a boot record, FAT, root directory, data area, and free space.
- The boot record (small program that runs upon starting the computer) determines whether or not the basic components for running the operating system are present.
- The FAT keeps lists of what is on the tracks and sectors.
- The operating system can find information on hard drives because each platter is labelled as tracks and sectors. (Think of sectors as the pieces of the pie (wedges) and the tracks as selection tracks on LPs)
- The root directory is the top folder (at the top of the file management system).
- Use the operating system to check out specific information on any file in a folder (e.g., filename, size, time, and date created or last modified).
- The data area is where programs are stored.
- The free area is what is left over to store files.

Appendix 3.1.2

Using the *Windows 95 Resource Kit* Help Files

See resources for other operating systems.

Purpose

Use the resource kit help files on the Windows 95 CD-ROM to recover the descriptions of “Disk Partitioning”, “Disk Formatting”, and “Scandisk”

Procedure

Place the Windows 95 CD into the CD-ROM drive.

Using windows explorer, go to directory “d:\admin\reskit\helpfile”

Double-click on the “win95rk” help file.

Select the “index” tab to perform your searches and record the descriptions of the following:

Partitioning Hard Disks

Format Utility

Scandisk Utility

Appendix 3.1.3

Types of Computers

Computer Type	Distinct Features	Typical Uses	Limitations/ Strengths	Setup Steps
Micro-controller				
PLC				
PC				
Workstation on Network				
Server				

Activity 2: Introduction to Networking Technologies

Time: 375 minutes

Description

Students review basic electronic circuits, computer connectivity, and the historical developments that generated the need for networking solutions. Students perform or observe a basic connectivity task involving two computers, a parallel or serial cable, and external devices (e.g., peripherals). Students also describe network types, topology, architecture, and cabling standards.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations

SPV.02E - demonstrate the use of an operating system, including a network;

ICV.01E - describe the evolution of computer electronics.

Specific Expectations

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;

TF2.02E - identify the basic internal and external components of a computer;

IC1.02E - use safe practices in the handling of computer hardware and electronic components;

IC1.03E - identify important scientific advances in computer electronic components;

IC1.08E - demonstrate understanding of the importance of ethical computer use.

Planning Notes

- Prepare for the construction of simple electronic circuits.
- Test the connectivity within the operating environment using the simplest method, either serial or parallel (e.g., Interlink in DOS using parallel ports).
- Prepare lesson presentation of topologies, architecture, network types, and cabling standards (Appendices 3.2.3 – Networks).
- Have various kinds of cable and connectors available for students to handle and connect, as well as stripped cable for student to see layers.
- Decide on alternative resources that would enhance this activity (e.g., guest speaker, brochures, field trip).
- Prepare an end of activity test of new terms and factual information (Appendix 3.2.4 – Networking Test).

Prior Knowledge Required

Students have:

- an understanding of the basic functions of the operating system. This would include an understanding of how to copy/delete/rename files, format data and bootable diskettes, and locate various hardware controls in the operating system;
- a clear understanding of the safe handling of computer hardware from Unit 1;
- use of the Safety Passport.

Teaching/Learning Strategies

(NB: suggested timings are approximate)

(75 min.)

- The teacher and students discuss concepts of connectivity from “SneakerNet” (copying files onto a disk then walking it to another computer) up to and including the Internet.
- Teachers and students identify other applications where computers must network (e.g., ATMs, heating and air conditioning systems, traffic lights, power control systems on a car, robotic manufacturing processes, etc.).

(75 min.)

- The teacher demonstrates how to connect two computers using a simple serial or parallel cable (e.g., ADB cable and the Chooser on a Macintosh). Students make a checklist during the demonstration and make a connection on their own computers using operating system utilities and transfer a file.
- The teacher and students discuss the impact on society from a technical and ethical perspective (e.g., ethical vs. unethical hacking) using role-play, demonstration, or other visual aids.
- The teacher introduces the report and accompanying rubric, provides resources for students to use to complete this task, and assigns due date (Appendix 3.2.1 – Positive and Negative Aspects of the Evolution of Connectivity). Students begin research on the report and continue for homework and/or time during class when other activities are complete.
- Students work as a team to connect two computers using a parallel/serial cable and appropriate operating system utilities. Using the checklist they created during the teacher demonstration, students check their machines for connection success and troubleshoot any difficulties. Circulate to assist students with any problems. Students complete their checklist to add to their portfolio and continue work on their report if time permits.

(75 min.)

- Students view *The Journey Inside The Computer* video Unit 6 on Networking (8 minutes and 45 seconds) with the purpose of reviewing computer components and introducing networks and operating systems. During the video, students record any new terminology that is unfamiliar to them. After the video the students and teacher clarify definitions (e.g., new terms might be: router, external bus, internal cache, bandwidth, gateway, host site, hub, node, repeater).
- Teachers and students review voltage, current, resistance, and Ohm’s Law from Grade 9 Science unit on electricity. Teachers and students review binary numbers and practise converting letters to numbers and short messages. (Hint – use 00000 as a blank, 00001 as an A, 00010 as a B, etc.)

(75 min.)

- In teams of three or four, students complete a hands-on activity that requires them to select appropriate conductors to build a simple serial communication circuit and a simple parallel communications circuit (Appendix 3.2.5 – Construction Activity). The teacher and students discuss the advantages and disadvantages of parallel and serial communication links.

(75 min.)

- The teacher presents the information on network types, topologies, architecture, and cabling standards (Appendix 3.2.3 – Networks). Students record this information in their notebooks for the end of activity test and label the diagram (Appendix 3.2.6 – Network Topologies).
- At the end of the activity the students do the Network Test (Appendix 3.2.4 – Networking Test). They may also continue work on their report.

Assessment/Evaluation Techniques

- Use Learning Skills Checklist from Unit 1 (Appendix 1.1.8 – Learning Skills Checklist).
- A formative assessment using problem-solving groups to address connectivity issues in the hands-on activity or described scenarios (the latter is in the absence of hardware). Teacher conferencing gives students feedback on progress and areas needed to improve.
- Assess student checklist of connectivity exercise.
- Summative evaluation of report using the report rubric (Appendix 3.2.2 – Report Rubric)
- Evaluate networking test using answers directly from student notes except for the last question which evaluates correct reasoning of information provided (Appendix 3.2.4 – Networking Test).

Accommodations

- Provide specific articles for research.
- Extend report deadlines.
- Provide a checklist for students during the connectivity activity.
- Create a poster on cabling uses and standards.

Resources

Software

Operating systems (e.g., DOS, Windows 9x, Windows NT, System 6.x, 7.x, 8.x, Unix, or others)

Print

Operating system manuals and reference texts

The Whole Internet: The Next Generation. O'Reilly and Associates Inc., 1999. ISBN 1-56592-428-2

Web Sites

<http://www.novell.com/catalog/primer/primer.html>

Video

Intel Corporation. *The Journey Inside*. Part of *The Journey Inside Education* kit

Appendix 3.2.1

Positive and Negative Aspects of the Evolution of Connectivity

Question Sheet

1. How and why did computer connectivity develop?
2. What are advantages and disadvantages of connectivity? Discuss using the following circumstances.
 - personal communication
 - business communication
 - collaborative processors
 - marketing
 - development of software
 - exchange of ideas
3. In your opinion, what is the greatest advantage/disadvantage of connectivity in an educational setting?

Appendix 3.2.2

Report Rubric

Criteria	Level 1 (50-60%)	Level 2 (60-70%)	Level 3 (70-80%)	Level 4 (80-100%)
Knowledge/Understanding				
Uses appropriate terminology	- occasionally uses appropriate terminology	- sometimes uses appropriate terminology	- mostly uses appropriate competent use of correct terminology	- always uses appropriate masterful use of correct terminology
Clearly states a choice	- developing indication of choice	- indication of choice	- successful indication of choice	- excellent indication of choice
Thinking/Inquiry				
Uses evidence to support recommendations	- emerging use of evidence to support recommendations	- adequate use of evidence to support recommendations	- successful use of evidence to support recommendations	- exemplary use of evidence to support recommendations
Communication				
Error-free word-processed report	- word processed with some errors	- word processed with few errors	- word processed with one or two errors	- word processed error-free
Application				
Extends to own and/or global societal issues	- limited awareness of own/societal issues	- adequate awareness of own/societal issues	- successful extension to own/societal issues	- expert extension to own/societal issues

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Appendix 3.2.3

Networks

Networks: Two or more communicating devices connected to form a system that shares applications, data, or hardware components.

Application of a Network

- Share resources.
- Speed connections between computers.
- Research collaboration.
- Electronic mail.
- Teleconferences and discussion groups.
- Co-operative Learning.

Local Area Networks (LANs)

- Computers are close together geographically.
- Computers connected to the network are called nodes.
- Computers and devices that allocate resources for a network are called servers.
- Cables connect the nodes together and transmit information between them:
 - twisted pair cable costs the least;
 - coaxial cable: can be baseband (can only carry one signal but is very fast) or broadband (can carry more than one signal);
 - fiber optic cable can carry massive amounts of data and is unaffected by magnetic or electrical interference, but is expensive and difficult to install.
- Cable interface unit (outside the computer) sends and receives signals on network cable.
- Network interface card (inside the computer) is connected to the cable interface unit by wire.
- Two networks can be connected using a bridge or gateway. If the computers are similar, use a bridge. If they are dissimilar, connect them with a gateway that will translate different data formats.

Topology

The physical layout of a LAN is known as topology. The most common layouts are:

- BUS - all nodes connected to LAN as branches on a common line;
 - requires network cards installed in nodes;
 - failure of one computer does not affect others on network.
- STAR - has a host computer, which is responsible for managing the network;
 - all nodes attached to separate lines which lead to hub;
 - if main computer fails - so does network.
- RING - all nodes on the same circuit forming a continuous loop with no host computer;

LAN Architecture

- Client/server networks use one or more computers as servers and other computers are clients.
 - Server contains the network operating system, software to run the network, application software and large data files.
 - Clients can request files, save files, run programs, and print on a network printer.
- On a peer-to-peer network all computers can access public files located on other computers on the network. Peer-to-peer architecture is used mainly on small networks, due to management problems.

Appendix 3.2.3 (Continued)

Networks

Network Communication

Application layer - what users see.

Presentation layer - ensures message is transmitted in a language that the receiving computer can read.

Session layer - opens communications and keeps all messages from various nodes straight.

Transport Layer - protects data being sent.

Network layer - selects route for message.

Data-link Layer - supervises the transmission.

Physical layer - encodes messages into the medium that will carry them (e.g., analog if message going by telephone line).

WAN – Wide Area Networks

- Computers are farther apart and connected by telephone lines or radio waves.
- Require special media provided by telephone companies.
- Require service providers.
- Require special hardware:
 - telephone wires, fiber optic cables, microwaves, or satellites for communication;
 - routers decide where to send data;
 - front end processors keep track of all messages and provide security.
- Used for EDI - Electronic Data Interchange.
- Internet is a very large example of a WAN.

Network Protocols

- A network protocol is a set of rules for exchanging information.
- Protocols assist in transferring files between microcomputers.
- When different formats meet it is called handshaking.
- Protocol converters translate information.
- TCP/IP is the most common protocol (Transmission Control Protocol/Internet Protocol).

Appendix 3.2.4

Networking Test

Name: _____

1. You have been hired to “walk me” through choices I could make in setting up a network. Under each heading explain my choice(s) and their differences. Be sure to use all your computer terminology in your explanation. (25 marks)
 - a) topology
 - b) architecture
 - c) network
 - d) cabling
 - e) network protocol
2. What is the purpose(s) of networking computers? (5 marks)
3. What problems arise from connectivity? (5 marks)
4. In your opinion, have advancements in network technology been largely positive or negative on the whole? State reasons for your choice. (5 marks)

Appendix 3.2.5

Construction Activity

Serial Communication Circuits

Materials Required

Wire cable – 3 to 4 metres in length (6 or 8 conductor)	Solder – 10 cm of SN63/Pb37
Marrette – several #29 and #31	Lamp holder - 4
Lamp – 4 (12 volt to fit lamp holder)	Switch – 4
Battery – small 12 volt gel cell (or a small laboratory power supply)	

Task

Using the above list of materials, student teams design, build (solder connections), and test the two serial communications systems (simplex and duplex) by sending assigned messages.

Method

1. Students design (showing proposed circuit schematic diagram to teacher for approval), then build a master station (with battery and switch) and slave station (with lamp) connected through the long cable, and finally demonstrate working series circuit.
2. Students design then add (after teacher approval), to the already built master station and slave station, a second series circuit, providing a timing light to simplify encoding and decoding serial messages, and demonstrate working series circuits.
3. Finally, students design and duplicate the already built master stations and slave station at the opposite ends of the cable and demonstrate sending binary message in two directions.

Parallel Communication Circuits

Materials Required

Wire cable – 3 to 4 metres in length (6 or 8 conductor)	Solder – 10 cm of SN63/Pb37
Marrette – several #29 and #31	Lamp holder - 5
Lamp – 5 (12 volt to fit lamp holder)	Switch – 5
Battery – small 12 volt gel cell (or a small laboratory power supply)	

Task

In small teams, students design, build (solder connections), and test a one-way parallel communications system by sending short binary encoded messages.

Method

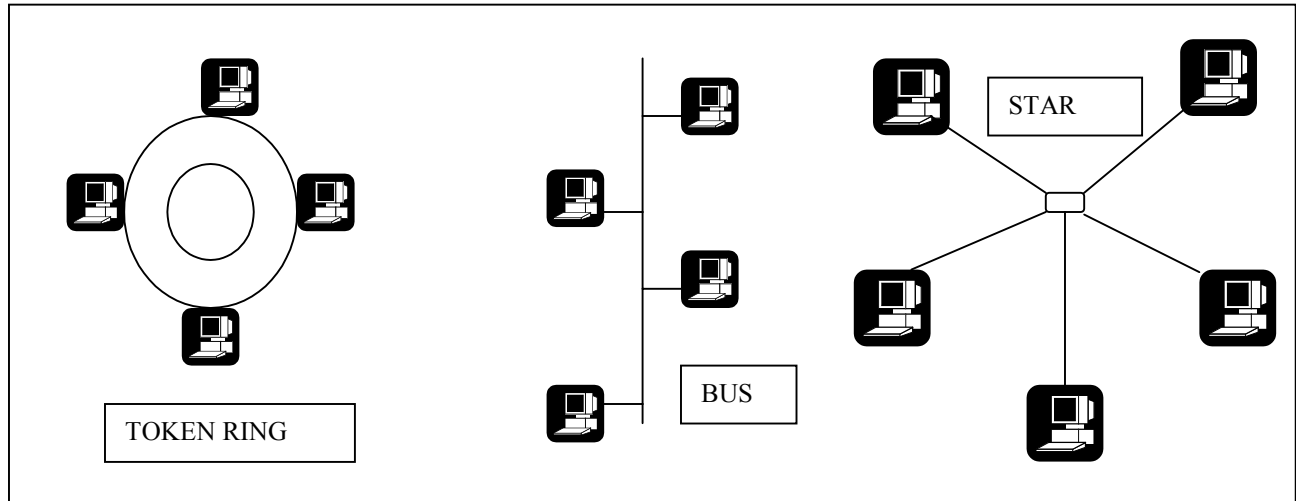
1. Students design (showing proposed circuit schematic diagram to teacher for approval), then build a sending station (with battery and five switches) and receiving station (with five lamps) connected through the long cable, and finally demonstrate working parallel communications circuit.

Conclusion

Comparison of Serial and Parallel Communications Circuits		
Characteristic	Serial Communication Circuits	Parallel Communication Circuits
Design	Requires a time signal	Very simple to understand
Speed to send information	Can be slow	Speed increases with more bits
Complexity	Few components	Can have many components
Cost over long distances	Low, only a few wires needed	High, many wires needed

Appendix 3.2.6

Network Topologies



Activity 3: Activating a Computer on a Network

Time: 150 minutes

Description

Students install a network interface card, activate it using the current operating system, and then connect it to another computer or network. The focus of this activity is hands-on manipulation of the hardware, which helps students understand physical aspects of network connectivity. Students also link the concepts of direct network connectivity to the Internet using a network interface card and dial-up networking using a modem.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations

TFV.01E - describe how the internal components of the computer enable the peripherals to function;
SPV.02E - demonstrate the use of an operating system, including a network.

Specific Expectations

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;
TF2.02E - identify the basic internal and external components of a computer;
SP2.05E - use correctly a variety of network system software;
SP2.06E - use Internetworking services correctly to access and navigate global information resources;
IC1.02E - use safe practices in the handling of computer hardware and electronic components;
ICI.09E - demonstrate compliance with acceptable use policies.

Planning Notes

- Test connectivity to determine if cables and network devices are in working order.
- This activity is best completed with older PCs that are not connected to the main school system.
- If a phone line is available, check connectivity (i.e., dial tone) and modem configuration on the demonstration computer.
- Prepare appropriate hardware to facilitate the installation of NICs (Network Interface Cards). A station approach can be used in a limited hardware situation. Older NICs are preferred for this activity as students may accidentally damage a card in learning the correct installation procedure.
- Review prepared step-by-step handouts and complete steps according to the work site (Appendix 3.3.1 – Installing a Network Interface Card Checklist).
- A networking technician may be invited to discuss issues associated with setting up and maintaining a network.

Prior Knowledge Required

Students have:

- a clear understanding of safe handling of computer hardware from Unit 1;
- an understanding of handling and use of network electronics and cabling from the previous activity.

Teaching/Learning Strategies

- Teachers and students review safety with static electricity and the importance of keeping contacts clean as they apply to components. Use the student Safety Passports (Appendix 1.1.2 – Student Safety Passports).
- Teachers introduce the activity with a discussion about connectivity to the Internet (the concepts of direct and dial-up connectivity are the focus). Students review handout (Appendix 3.3.1 – Installing a Network Interface Card Checklist) and follow the checklist as the teacher demonstrates:
 - the installation of a NIC;
 - the installation of a driver to activate the NIC;
 - illustrate link status indicators to show that the NIC was successfully activated;
 - demonstrate the installation of a driver;
 - installation of an appropriate network protocol;
 - connection to a network and how to test for successful connectivity.
- In groups students refer to Appendix 3.3.1 and install a NIC. The teacher assists with any problems that arise. Students are required to fill in terminal logs to record any changes made to the computers.
- Students and teacher discuss the direct connection to the demonstration of dial-up networking via a modem. Students illustrate in their journals the similarities and differences between direct connection and how this form of connectivity is evolving. Students complete the worksheet (Appendix 3.3.2 – Evolution of Connectivity).

Assessment/Evaluation Techniques

- Learning Skills Checklist (Appendix 1.1.8)
- Provide verbal feedback to students to assist them in successful completion of the task. This task is best assessed using a checklist (see Unit 1).

Accommodations

- Provide one-to-one teaching.
- Peer helpers would be an asset.
- Provide step-by-step instructions and/or a glossary of terms.

Resources

Software

Operating systems (e.g., DOS, Windows 9x, Windows NT, System 6.x, 7.x, 8.x, Unix, or others)

Print

Operating system manuals and reference texts

Network+ Certification Study Guide. Syngress Media, Inc., 1999. ISBN 0-07-211846-6

Web Sites

Novel Primer

<http://www.novell.com/catalog/primer/primer.html>

Appendix 3.3.1

Installing a Network Interface Card Checklist

Names of Team Members:		
	Instructions	Completed
1.	Check to see if all appropriate hardware and tools are ready.	
2.	Attach a ground cable to your wrist and ground yourself to the computer chassis.	
3.	Open the computer chassis.	
4.	Check for open slots.	
5.	Identify the slot you are going to use for the NIC.	
6.	Remove the slot cover.	
7.	Retrieve the appropriate NIC for the slot you selected.	
8.	Put the NIC in the slot by putting the back end in first and then slowly rock the card forward.	
9.	Make sure your fingers are on the outside of the case and that the card faceplate does not catch the end of the case.	
10.	Firmly press the card in place then screw it in.	
11.	Close the case.	
12.	Remove your ground wire	

Software configuration instructions beyond this point vary significantly depending on the operating system, but, generally, if you restart your system the operating system prompts you for input commands to configure the device.

Appendix 3.3.2

The Evolution of Connectivity

Type of Connection	Speed	Evolution	How does this affect us?
Modem			
ISDN			
Cable			
ADSL/HDSL			
Direct LAN			

Activity 4: Network Options and Security

Time: 300 minutes

Description

The focus of this unit is the options for network users (clients) and security considerations on a network. Students learn how hardware and software resources may be shared by clients and how to install additional software on a server. Then students use the series and parallel circuits completed in the previous activity to communicate signals. Students also learn how the evolution of computer engineering has affected people's lives through password security issues and security on a network.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations

SPV.02E - demonstrate the use of an operating system, including a network;

ICV.02E - identify the social impact of computers and associated technologies.

Specific Expectations

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;

SP2.05E - use correctly a variety of network system software;

SP2.06E - use Internetworking services correctly to access and navigate global information resources;

IC1.02E - use safe practices in the handling of computer hardware and electronic components;

IC1.07.E - describe how computer engineering has evolved and how it has affected people's security, safety, and privacy.

Planning Notes

- Split the class into two groups: one group performs the hands-on activity of connecting the second printer and loading a network software application, the other group works on security issues to allow for sharing of hardware resources.
- Inventory the hardware available for any hands-on or demonstrative activities. Determine if appropriate quantities are available.
- Test connectivity between devices to determine if cables and network electronics are in working order.
- Test computer configuration to determine if all computers are operating correctly.
- Have appropriate network application software available.

Prior Knowledge Required

Students have:

- clear understanding of the safe handling of computer hardware;
- an understanding of the handling and use of network electronics and cabling from the previous activity;
- an understanding of the installation of software from Unit 1 and the previous activity;
- understand the topology of the network.

Teaching/Learning Strategies

- Teachers and students discuss applications where multiple users require access to the same information stored in a central database (e.g., ATM, libraries). Teachers and students discuss how networks can have shared hardware resources (e.g., printer).

-
- Teachers and students discuss why it is desirable to have more than one printer choice (output device) and the choice of several software applications. Also, the class develops ideas they feel contribute to security problems on networks not associated with stand-alone PCs. (passwords, encryption, software piracy).
 - In groups of two or three, students choose a security issue to research on the web (Appendix 3.4.1 - Security), using web search tools and sites provided in Resources. They prepare a short synopsis of their issue and deliver an oral report to the rest of the class using the rubric as their guide (Appendix 3.4.2 - Oral Report Rubric).
 - Small groups of students take turns setting up a second printer under the direction of the teacher and loading one network application, while the rest prepare their oral report or work through the second hands-on activity (Appendix 3.4.3 – Communicating at a Distance).
 - Once all groups have completed the hands-on activities, students access the new software to ensure success.
 - During the final day, students present their report to the rest of the class.

Assessment/Evaluation Techniques

Assessment and evaluation of the oral report using the rubric (Appendix 3.4.2 – Oral Report Rubric)

Accommodations

- Allow for written reports or taped reports instead of oral.
- Allow flexible timelines for report due date.
- Provide written articles as an alternative to web research.
- As enrichment, students may investigate a single transistor switch.

Resources

Software

Operating systems (e.g., DOS, Windows 9x, Windows NT, System 6.x, 7.x, 8.x, Unix, or others)

Web and FTP server and client applications

Print

Network+ Certification Study Guide. Syngress Media, Inc., 1999. ISBN 0-07-211846-6

Operating system manuals and reference texts

The Whole Internet: The Next Generation. O'Reilly and Associates Inc., 1999. ISBN 1-56592-428-2

Web Sites

Novel primer

<http://www.novell.com/catalog/primer/primer.html>

Security – Protecting Your PC and Media Against Computer Viruses

www.utoronto.ca/security/virus.htm

Computer Viruses – popular viruses and prevention

www.ccs.queensu.ca/pubs/itsnote/virus.html

Government of Canada, The Privacy Act – intellectual property protection

<http://www.privcom.gc.ca/privacyact1/htm>

Data Security – Information on password security

<http://www.cown.edu.au/ITDivision/security/password.htm>

Encryption – Security issues around encryption

www.symantec.com/avcenter/security/encryption/encrytion.html

Appendix 3.4.1

Security

Topics: (suggested only)	Definition
Shareware	
Privacy	
Software piracy	
Copy protection	
Site licensing	
Viruses	
Hacking	
Worms	
Trojan horse	
Password Security	
Encryption	
Spamming	
Internet theft	
Net nanny	
Firewalls	
Intellectual Property laws	
SET (Secure Electronic Transactions)	
Data backups	
Hard disk disaster recovery	
Vaccines	

Appendix 3.4.2

Oral Report Rubric

Criteria	Level 1 (50-60%)	Level 2 (60-70%)	Level 3 (70-80%)	Level 4 (80-100%)
Knowledge and Understanding Information complete	- provides some information on topic	- provides most information on topic	- provides complete information on topic	- all information is relevant and on topic
Thinking/ Inquiry Organization	- some content is well planned and organized	- most content is well planned and organized	- logical and easy to follow	- audience can anticipate next step
Communication Clarity of information	- limited clarity of information	- moderate clarity of information	- considerable clarity of information	- clear concise information
Uses appropriate terminology	- some terminology used effectively	- most terminology used effectively	- terminology used effectively	- terminology used exceptionally well
Confidence in topic	- limited command of topic	- unsure of content at times	- confident in topic content	- confident in topic and included presentation skills

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Appendix 3.4.3

Communicating at a Distance

Students have already constructed series and parallel circuits. Using their circuits, students make any refinements necessary to compete in a message sending and decoding challenge. The challenge for students is to send messages across the room to another team. The other team receives the message accurately and successfully decodes the message. (Prizes may be given to successful teams).

Activity 5: Networking

Time: 225 minutes

Description

A guest speaker is invited to share networking in a large environment with the students. Students expand on their computer industry knowledge from Unit 1 to include national and global perspectives. Students look at degrees and certifications available and opportunities they have at the high school level and beyond to move them toward careers in computer engineering.

Strand(s) and Expectations

Strand(s): Impact and Consequences

Overall Expectations

ICV.02E - identify the social impact of computers and associated technologies;

ICV.03E - identify related computer careers.

Specific Expectations

IC1.05E - describe careers related to computer engineering;

IC1.10E - identify computer skills that are important to employers.

Planning Notes

- Guest speakers may include the school sysop, board technician, or someone from the local community.
- Collect information in conjunction with the Student Services Department including school course calendars and college/university catalogues.
- Gather copies of recent computer trade magazines (especially Canadian sources).
- Arrange ahead of time for a student to introduce guest speakers and another student to thank them.
- Collect newspaper advertisements for jobs in the computer industry.
- Distribute a sample certification worksheet (Appendix 3.5.1 – Computer Engineering Certification) electronically if resources allow.

Prior Knowledge Required

Word-processing skills

Teaching/Learning Strategies

- Teachers introduce the expectations of the activity.
- Teachers review with students (ahead of time) questioning techniques for the guest speaker.
- One student may introduce the guest speaker. Students take brief notes in order to ask relevant and interesting questions. One student may thank the guest speaker.
- Discuss the speaker information with the students, after which they write their personal views on the information.
- Students expand on the career information they gathered in Unit 1 to include a more national focus. From articles in computer trade magazines, they are able to see advances that Canada is making in the computer industry. Each student chooses one article to word process a review of using the rubric provided to assess their progress. (Appendix 3.5.2 – One-Page Report Rubric).

-
- Finally, students look at opportunities for different computer designations ranging from MCSE (Microsoft Certified Engineer) to computer engineering at the university level. Students use newspaper advertisements to explore what skills and designations are requested by potential employers. Students compare the skills requested to their skill inventory completed in Unit 1 and stored in their portfolio. They may wish to revise their personal goals.
 - Students retrieve the certification chart file (either electronically or via handout) and, using designations discovered in the advertisement exercise above, they complete the chart and add it to their portfolio.
 - Students create a plan on how to pursue one of these designations, beginning with the completion of this course, and save the information in their portfolio (long-term goal).

Assessment/Evaluation Techniques

- Learning skills checklist (Appendix 1.1.8)
- Review of student portfolio to provide written/oral feedback on completion and comprehension of tasks given
- Evaluate the article review using the rubric provided (Appendix 3.5.2 – One-Page Report Rubric).

Accommodations

- Allow flexible timelines for due date of report.
- Use career centre videos if available.
- Invite the Student Services resource personnel into the classroom.
- Videotape the guest speaker(s) presentation to allow students an opportunity to watch it again.

Resources

Software

Ontario Educational Software Services (OESS) (e.g., Corel *WordPerfect*, *Microsoft Works*, *Appleworks*, etc.) Note: OESS is part of the TVO Teacher Services.

Print

The Whole Internet: The Next Generation. O'Reilly and Associates Inc., 1999. ISBN 1-56592-428-2

Web Sites

Novell Primer

<http://www.novell.com/catalog/primer/primer.html>

IT Careers – careers database

<http://www.itcareers.com>

Globe and Mail: Canada's National Career Site – careers database

www.globecareers.com

Canada Employment Search Network – careers database

www.canjobs.com

Canadian Career Page - careers database

www.canadiancareers.com

Global Knowledge – IT training (Canadian source)

www.am.globalknowledge.com

Appendix 3.5.1

Computer Engineering Certifications

Certification	Description and level of acceptance	Career potential with Certificate
e.g., MCSE	Microsoft Certified Systems Engineer – requires six completed courses. Based on IT job requirements. Accepted by many companies.	Career possibilities as a result of certification is high.
CISCO	CISCO Systems Certified	
3COM	3COM Systems Certified	

(if possible this chart should be distributed electronically so that information can be typed into the cells in a table, otherwise more space is to be provided for writing)

Appendix 3.5.2

One-Page Report Rubric

Criteria	Level 1 (50–60%)	Level 2 (60-70%)	Level 3 (70-80%)	Level 4 (80-100%)
Knowledge/Understanding				
Uses appropriate terminology	- emerging use of appropriate terminology	- adequate use of appropriate terminology	- competent use of appropriate terminology	- masterful use of appropriate terminology
Clearly states a choice	- developing indication of choice	- adequate indication of choice	- successful indication of choice	- excellent indication of choice
Thinking/Inquiry				
Uses evidence to support recommendations	- emerging use of evidence to support recommendations	- adequate use of evidence to support recommendations	- successful use of evidence to support recommendations	- exemplary use of evidence to support recommendations
Communication				
Error-free word-processed report	- word processed with several errors	- word processed with few errors	- word processed with one or two errors	- word processed with no errors
Application				
extends to own and/or global societal issues	- awareness of own/societal issues	- adequate awareness of own/societal issues	- successful extension to own/societal issues	- expert extension to own/societal issues

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Unit 4: Computer Programming

Time: 27 hours

Unit Description

This unit focusses on how to program a computer using a problem-solving model. This model helps to organize and develop the fundamental structures of programming. These fundamental structures include variable declarations, assignment statements, input/output structures, selection, and looping. Each structure builds upon and is incorporated into subsequent structures. The programming software introduced in this unit allows students to apply simple programs that integrate with hardware to control external devices and peripherals. Students also research and identify computer programming related careers and explore ergonomics.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Process, Impact and Consequences

Overall Expectations: TFV.02E, TFV.03E, TFV.05E, SPV.02E, SPV.04E, ICV.01E, ICV.03E.

Specific Expectations: TF1.01E, TF1.02E, TF3.01E, TF3.02E, SP2.03E, SP2.04E, SP3.01, SP3.02E, SP3.03E, IC1.01E, IC1.04E, IC1.05E, IC1.06E, IC1.07E, IC1.09E, IC1.10E.

Activity Titles (Time + Sequence)

Activity 1	Programming Environment and Related Careers	330 minutes
Activity 2	Input/Output Structures	450 minutes
Activity 3	Repetition Structures	420 minutes
Activity 4	Decision Structures	420 minutes

Prior Knowledge Required

All students know the order of operations in math and file management skills from earlier units.

Unit Planning Notes

Students work in a computer environment in a safe and healthy manner by learning about potential problems associated with extended computer use. The main focus of the unit is to introduce students to computer programming and its appropriate ethical use. Students incorporate a computer problem-solving model to design and write computer programs. The programming structures necessary to write a program are introduced individually but build on previous structures. The programming language used depends upon what software and hardware are available in the school. Several programming structures are provided including how to write a program capable of controlling the mouse and parallel port for the next unit. Refer to the Course Profile for Computer Information Science for additional programming activities. The initial activity examines the changes in the engineering profession and would be best approached by having a guest speaker from the computer engineering field. In the final activity, students are given a program to trace the input, processing, output model and see how an external device is controlled by the PC. Note there is an appendix to give you the description and planning notes if you wish to demonstrate this process with micro-controllers (Appendix 4.5.1 – Using Micro-controllers).

Teaching/Learning Strategies

Teaching and learning strategies include brainstorming, peer practice and teaching, research, process notes, hands-on computer programming activities, and Socratic lessons. Experienced students acting as peer guides and tutors could provide assistance with programming concepts.

Assessment and Evaluation

Some students have prior computer programming knowledge and skills. Formative assessment of the first few programming challenges may help identify knowledge and skill levels and allow teachers to make sure that all students are challenged. A mixture of assessment strategies including assignments, quizzes, tests, research, and checklists are used to gather assessment information towards a summative unit evaluation. The rubric (Appendix 4.0 – Programming Rubric) may be used for assessment/evaluation of programming in Activities 2, 3, and 4. Teachers may also use the Learning Skills Checklist found in Unit 1 for ongoing assessment.

Resources

Software

Computer programming software (possibly the same software used in Computer and Information Science)

Print

Carter, John. *Problem Solving in Pascal*. Toronto: Addison-Wesley, 1989. ISBN 0-20111215-9

Elgarten, Gerald, Alfred Posamentors and Stephens Moresch. *Using Computers in Mathematics*. Toronto: Addison-Wesley, 1983. ISBN 0-201-10450-4

Hume, J.N.P. *Problem solving and Programming in Turing*. Toronto: Holt Software Associates Inc., 1993. ISBN 0-921598-19-X

Web Sites

Exercises for Health and Safety at work – ergonomic exercises

<http://www.ergaerobics.com/ergercises.htm>

Workplace tools for Health and Safety – helpful suggestions for staying Healthy on the Job

<http://ergohealth.net>

Ergonomics – ergonomic exercises and equipment

<http://ergonomics.ucla.edu/Ergowebv2.0>

QBasic: Utilities – basic programming information

<http://qbutil.hypermart.net/>

Career and Education – career and education database

<http://www.diversitycareers.com>

The Toronto Star – career database

<http://www.thestar.com>

The Globe and Mail – search engine to careers and various topics

<http://www.globeandmail.com>

The National Post – search engine to careers and various topics

<http://www.nationalpost.com>

Appendix 4.0

Programming Rubric

Criteria	Level 1 (50-59%)	Level 2 (60-69%)	Level 3 (70-79%)	Level 4 (80-100%)
Knowledge and Understanding				
Follows computer problem-solving model	- follows some aspects of computer problem-solving model	- follows most aspects of computer problem-solving model	- follows computer problem-solving model	- uses computer problem-solving model in a variety of contexts
Followed style guide	- some of the style guide was followed	- most of the style guide was followed	- the style guide was followed	- the style guide was followed precisely
Thinking/Inquiry				
Solving the problem	- identifies some of the problem and requirements	- identifies most of the problem and requirements	- solves the problem with user requirements or satisfaction	- solves the problem to full user requirements and satisfaction; clarity and insight at each stage
Program Runs	- program has trouble at most stages	- program runs at some stages	- program runs with few problems	- program runs smoothly from beginning to end
Application				
Documentation and Design conventions	- partially complete with ambiguities and often does not adhere to program design conventions	- complete with few ambiguities and sometimes does not adhere to program design conventions	- complete, clear, and adheres to program design conventions	- complete, clear, and adheres precisely to the specific program and design conventions

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Activity 1: Programming Environment and Related Careers

Time: 330 minutes

Description

Students discover how to avoid any health or safety issues related to computer use. Students learn a problem-solving approach to programming using written solutions. Students are then introduced to the computer-programming environment, which includes how to retrieve, edit, and run programs. Students explore how careers in engineering fields are integrated with many professional fields of study. They explore the impact of software for specific uses within these professions. Finally, students identify education and skills required by employers by examining various job advertisements for computer engineers within various professions.

Strand(s) and Expectations

Strand(s): Skills and Processes, Impact and Consequences

Overall Expectations

SPV.02E - demonstrate the use of an operating system, including a network;

ICV.03E - identify related computer careers.

Specific Expectations

SP2.03E - trace the operation of a system consisting of a program, an interface, hardware, and directories;

SP2.04E - use appropriate file management techniques;

IC1.01E - use appropriate strategies to avoid potential health and safety problems associated with computer use, such as posture problems, eye strain, and musculoskeletal injuries;

IC1.04E - describe the development of computer engineering technology and its impact;

IC1.05E - describe careers related to computer engineering;

IC1.06E - analyse the influences of computers on the engineering profession;

IC1.07E - describe how computer engineering has evolved and how it has affected people's security, safety, and privacy;

IC1.10E - identify computer skills that are important to employers.

Planning Notes

- Plan ergonomic exercises that work for your room area (Appendix 4.1.1 – Ergonomic Exercises) and make an overhead of the workplace assessment sheet (Appendix 4.1.2 – Workplace Ergonomic Assessment).
- Distribute copies electronically of Appendix 4.1.3 – Problem-solving Strategy and a first program to run and edit (Appendix 4.1.4 – Early Program). Note the samples used in this unit are written using Qbasic. If using Macintosh, refer to Resources for the web site to download the free Basic interpreter called Chipmunk BASIC. If you are using the LINUX operating system, the resource list includes the web site to download the free BASIC interpreter called YABASIC (Yet Another BASIC). Furthermore, you may wish to consult the Computer and Information Science profile for examples of the Turing programming language.
- Identify an appropriate storage area for student programs and suggest appropriate backup practice (e.g., Disk, alternative drive).
- Language selection considerations should include two additional requirements for Unit 5, namely controlling the mouse and accessing the parallel port through software.
- Consideration must also be given that the operating system software does not conflict with the programming software when accessing the parallel port.

-
- Adapt checklist from Unit 1 (Appendix 1.1.7 – Activity Checklist).
 - Organize career and computer programming reference material. Check any Internet sites to ensure availability.
 - An alternative to a research assignment could include student interviews of a computer engineer or guest speakers from the computer engineering field with a professional twist (e.g., someone who programs for medical equipment).

Prior Knowledge Required

Students have an understanding of file management and keyboarding skills. Students also require research skills.

Teaching/Learning Strategies

- Begin the lesson with a question to students about what they would improve in the classroom. Choose suggestions that relate to health and safety concerns to explain to students causes of posture problems, eye strain, and musculoskeletal injuries.
- Using the overhead of the workplace assessment (Appendix 4.1.2 – Workplace Ergonomic Assessment), come to a class consensus of the classroom as a healthy and safe work environment. Request solutions to those problems that can be readily remedied. (You may want students to assess their home environment for homework).
- Lead the students through a series of simple exercises they can do on the job to relieve tension and reduce the likelihood of any injuries from extended computer use (Appendix 4.1.1 – Ergonomic Exercises).
- To bring the students back into focus, start the problem-solving portion with a brain teaser. E.g., Bill is five years older than his sister now but at one time he was twice her age. How old was his sister when he was twice as old?
- Ask for suggested solutions and accept all answers. Use the problem-solving model (Appendix 4.1.3 – Problem-solving Strategy) to work through the problem in a logical manner.
- Introduce a second problem (e.g., If Joseph is three times as old as Jacob will be when Jerry is as old as Joseph is now, what is the order in age from oldest to youngest?) that students do on their own and then ask for volunteers to present the problem using the problem-solving strategy.
- Ask students to logon to the system and retrieve the file containing the problem-solving strategy. Students should have a copy of this (electronically or in print) stored for easy reference.
- Demonstrate the programming software and walk all students through the process of retrieving a program and running it. (A palette or projector is helpful.) Finally, students retrieve a program to edit, then resave as their first program (Appendix 4.1.4 – Early Program).
- Introduce students to the idea that programming is not solely related to games, but to controlling devices through the use of a computer. Explain how peripherals attached to computers require instructions that allow them to complete their task (e.g., sensors, robotic arms). For this reason, many professions require the use of good programmers who have knowledge in their field of study (e.g., computerized geophysical equipment benefits from a programmer who is well versed in the field of geophysics).
- Provide students with a project checklist and assessment rubric (Appendix 4.1.5 – Career Assignment Checklist and Appendix 4.1.6 – Professions and Computer Engineering Assessment Rubric).
- Students choose a field of study (e.g., manufacturing, medicine, architecture, business, geology, mining, etc.). Using job search web sites, newspapers, and/or personal contacts, students research ways in which computer engineers are being employed in that field of study. Circulate to assist students in locating appropriate material.
- Students present their findings on a poster, in a newsletter, or other visual format to create a classroom display. Post all final products to allow all students to expand their knowledge in this area.

Assessment/Evaluation Techniques

- Learning skills checklist
- Use checklist from Unit 1 adapted for Unit 4 to indicate ability to retrieve, edit, and resave programs.
- Watch for students using the exercises and encourage their use whenever possible.
- Summative assessment of material using presentation rubric (Appendix 4.1.6 – Professions and Computer Engineering Assessment Rubric)

Accommodations

- Assist students filtering information when needed.
- Make arrangements with resource staff for students requiring more assistance than can be given in a regular classroom setting.
- Dialogue/conference with students to discuss accommodations to make certain that the present surroundings meet their needs.
- Students with programming experience from Integrated Technologies, Grade 9, could assist other students.
- Create posters of ergonomic exercises.
- Students may create small programs to use in future lessons.
- Pair students for this activity.
- Assist students filtering information as necessary.
- Make arrangements with resource staff for students to visit the career centre if available.
- Allow alternative presentation formats.

Resources

Software

Computer programming software (possibly the same software used in Computer and Information Science)

Print

Carter, John. *Problem Solving in Pascal*. Toronto: Addison-Wesley, 1989. ISBN 0-20111215-9

Elgarten, Gerald, Alfred Posamentors, and Stephens Moresch. *Using Computers in Mathematics*. Toronto: Addison-Wesley, 1983. ISBN 0-201-10450-4

Web Sites

Chipmunk BASIC – free download of BASIC interpreter for Macintosh users

<http://www.nicholson.com/rhn/basic>

YABASIC – free download of BASIC interpreter for LINUX operating systems

<http://www.yabasic.de>

Appendix 4.1.1

Ergonomic Exercises

1. **Wrists:** to increase circulation and stretch wrist muscles.
Hold right arm out at shoulder height.
Flip your right fingers up so they are pointing at the ceiling.
With your left hand, gently pull back on your right hand fingers.
Hold for 10 seconds and repeat five times.

Reverse Hands.
2. **Posture:** Exercise to strengthen the muscles in your neck, back, and stomach to help hold yourself up.
Neck rolls - repeat five times each way
Stretches – stretch your hands over your head , hold for three seconds and repeat five times.
Back – standing, place your hands on your hips and lean back to look at the ceiling and return to standing position. Repeat ten times.
Back and shoulders – sitting in your chair, reach across body and grab the back of the chair. Pull gently and hold for five seconds. Reverse sides.

Appendix 4.1.2

Workplace Ergonomic Assessment

Item	Rating 1 is not ergonomic and 10 is ergonomic	Improvements Suggested
Lighting	1 2 3 4 5 6 7 8 9 10	
Air Quality	1 2 3 4 5 6 7 8 9 10	
Temperature	1 2 3 4 5 6 7 8 9 10	
Chairs	1 2 3 4 5 6 7 8 9 10	
Desks/tables	1 2 3 4 5 6 7 8 9 10	
Keyboard height	1 2 3 4 5 6 7 8 9 10	
Noise	1 2 3 4 5 6 7 8 9 10	
Monitors	1 2 3 4 5 6 7 8 9 10	
Keyboards	1 2 3 4 5 6 7 8 9 10	
Mice	1 2 3 4 5 6 7 8 9 10	

Adapted from the BTT Grade 9 Course Profile

Appendix 4.1.3

Problem-Solving Strategy

1. Define the Problem: What needs to be solved?
2. Analyse the Problem: What are the facts? What do I assume? What else do I need to know?
3. Possible Solutions: List all possible solutions.
4. Prioritize the Ideas: Establish the pros and cons of possible solutions, and rank them.
5. What steps do I need to take to get to the top ranked solution?
6. Evaluate: Does this solve my problem?

NB: Sometimes you are solving a problem for someone else. In this case you need to consider their needs in Step 2 (e.g., language level, ability) and again in Step 6.

Appendix 4.1.4

Early Program

```
REM First Program
REM Teacher
REM January 2000
REM This is a first input/output program
CLS
PRINT "What is your first name?"; Name$
INPUT Name$
CLS
PRINT
PRINT Tab(21); Name$; ", welcome to the programming portion"
PRINT Tab(21); "of this course. I hope you enjoy it!"
END
```

Appendix 4.1.5

Career Assignment Checklist

Name:

Teacher:

Due Date:

You are learning about computer programs. With this short practical introduction, you will have a feeling of what a programmer does and possible careers related to programming. The purpose of this assignment is to relate the information to the variety of professional fields in which programming knowledge and skills would be useful.

INCLUDE:

Professional field

List programming positions in this field (include your source of information).

For each position include:

- company name;
- programming environments listed;
- salary range (if given);
- educational requirements to pursue this position;
- computer skills required;
- other skills required;
- programming languages required in advertisement.

Compare high and low level programming languages (if applicable in career).

Arrange all information attractively in a visual presentation for the class.

Include a bibliography.

Appendix 4.1.6

Professions and Computer Engineering Assessment Rubric

Criteria	Level 1 (50-59%)	Level 2 (60-69%)	Level 3 (70-79%)	Level 4 (80-100%)
Knowledge/Understanding				
Uses correct terminology	- limited use of correct terminology; limited knowledge of social or environmental impact	- adequate use of correct terminology; some knowledge of social or environmental impact	- competent use of correct terminology; considerable knowledge of social or environmental impact	- masterful use of correct terminology; exemplary knowledge of social or environmental impact
Thinking/Inquiry				
Uses evidence to support rationale	- some use of evidence to support rationale	- general use of evidence to support rationale	- effective use of evidence to support rationale	- a variety of evidence to support rationale
Communication				
Choice of presentation medium	- some planning in presentation medium	- presentation medium is adequate for topic	- presentation medium suits topic	- presentation medium enhances topic and is memorable
Accurate/varied bibliography	- some formatting present	- mostly formatted bibliography	- bibliography format is correct and has adequate variety	- bibliography is complete in depth and variety
Application				
Makes a connection to purpose	- some indication of purpose	- indication of purpose	- successful connection to purpose	- unique and effective indication of purpose

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Activity 2: Input/Output Structures

Time: 450 minutes

Description

Students apply a computer problem-solving model to write and to document programs involving input and output structures. The three fundamental variable types (real, integer, and string) are used in the input and output structures. Students are given a style guide to use as a checklist when solving all problems using programming.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Process

Overall Expectations

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.03E - explain internal numbering and character representation systems;

TFV.05E - describe the fundamental programming constructs;

SPV.04E - apply fundamental programming constructs to develop programs that interact with external components.

Specific Expectations

TF1.01E - describe the relationship between the binary number system and computer logic;

TF1.02E - define a standard way of representing characters in a binary code;

TF3.01E - define constants, variables, expressions, and assignments statements, including the order in which the operations are performed;

TF3.02E - describe how computers store and work with different types of data, including numbers and characters;

SP3.01E - use input and output statements in a program.

Planning Notes

- Prepare copies of style guide with the sample programs on the back (Appendix 4.2.1 – Programming Style Guide and Appendix 4.2.2 – Sample Programs).

Prior Knowledge Required

Students require a thorough understanding of the programming portion of Activity 1.

Teaching Learning Strategies

- Thoroughly review the programming style guide with students (Appendix 4.2.1 – Programming Style Guide). Encourage good programming practices from the beginning (e.g., Use Save As command first, use problem-solving method, follow Style Guide).
- Students open their programming editor and enter the sample program using correct form. Students run this program. If they encounter difficulties (as some will), instruct students to begin tracing to detect any errors they have made in data entry. Explain to students the difference between the two types of errors: syntax and logic (Appendix 4.2.2 – Sample Programs).
- Review the programming process with students. 1. Use a problem-solving strategy and outline on paper. 2. Use the style guide to assist with program planning. 3. Code program, run, and debug if necessary.
- Explain to students the three types of variables (real, integer, and string). Also include remarks on reserved words, assignment statements, keyboard input, and spacing of output.

-
- Review the programming rubric with students to use as a self-assessment tool while working on individual programs. Students work through a series of small problems using the programming process taught in this activity. Students submit their best two problems for evaluation (Appendix 4.2.3 – Problem Sheet No. 1).

Assessment/Evaluation Techniques

- Formative assessment on problems giving written/oral feedback to students
- Summative programming assignment on two problems presented by students using the Programming Rubric (Appendix 4.0 – Programming Rubric)

Accommodations

- Students may select their own problems to solve.
- Pair students to test each other's programs to provide assessment.
- Provide visuals of input/output examples.
- Allow varied timelines for completion of problems.

Resources

Software

Current Ministry of Education licensed computer programming software (possibly the same software used in Computer and Information Science)

Print

Carter, John. *Problem Solving in Pascal*. Toronto: Addison-Wesley, 1989. ISBN 0-20111215-9

Elgarten, Gerald, Alfred Posamentors, and Stephens Moresch. *Using Computers in Mathematics*. Toronto: Addison-Wesley, 1983. ISBN 0-201-10450-4

Appendix 4.2.1

Programming Style Guide

Header

The header is always written at the top of the program and must include:

- the program name
- the author's name (you)
- the date the program was created
- any modification dates
- description of the program.

Comments

- Each variable, constant, and array must be declared.
- Initialize variables.
- Comment at the top and bottom of loops (for/end for) and each conditional statement (if then/else/end if).
- Each procedure or function must indicate where the main program starts and ends.

Structure

Header

Declare local variables/arrays.

Initialize local variables.

Main Program starts here

Main Program ends here

Indentation Rules

Programs must be indented properly. All code:

- inside each loop must be indented;
- in conditional statements must be indented.

This style guide must be followed for every program you write.

Appendix 4.2.2

Sample Programs

```
REM Getting Input
REM Teacher
REM January, 2000
REM To get user input
CLS
PRINT "What mark would you like in this unit?"
PRINT
INPUT mymark$
CLS
PRINT "I would like to receive a " mymark$ " on this unit."
END
```

```
REM Print Choices
REM Teacher
REM January 2000
REM Demonstrate print commands
CLS
PRINT "Output can be fun!"
PRINT
PRINT "You can print numbers like"; 98.5; "."
PRINT
PRINT "I can"; " help you"; " learn the"; " difference"; " among"
PRINT
PRINT "using"; Spc(6); "the"; Spc(6); "SPC"; Spc(6); "command,"
PRINT
PRINT "using"; Tab(20); "the"; Tab(35); "TAB"; Tab(48); "command,"
PRINT
PRINT "using"; " semi-colons,"; " and"
PRINT
PRINT "using", "a", "comma", "to", "space."
END
```

Appendix 4.2.3

Problem Sheet No. 1

1. Design and write an output program that enters the following information on successive lines.
 - the number five;
 - the variable radius;
 - the word summer;
 - the number 3.14159;
 - the variable width followed by the units in cm;
 - the variable area of a circle in square metres.
2. Design and write a program that allows a user to input their first, middle, and last name and displays a greeting followed by their name (e.g., Hi Bill Edward Smith).
3. Design and write a program that allows a user to input the names of three employees, their rate of pay, and their number of hours, and displays this information in a logical manner. Be sure to explore pleasing spacing for your output.
4. Develop a standard header for your programs. Save this information and include it in each of your programs.
5. Design and write a program that demonstrates the various forms of input and output you have learned (e.g., words, numbers (long, short decimals), spacing conventions).

Activity 3: Repetition Structures

Time: 420 minutes

Description

Students apply a computer problem-solving model to create programs that include repetition structures. These programs incorporate the programming environment from Activity 1 and input/output and variable types from Activity 2. This activity ends with a formal assessment activity of programming concepts learned to date.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skill and Process

Overall Expectations

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.03E - explain internal numbering and character representation systems;

TFV.05E - describe the fundamental programming constructs;

SPV.04E - apply fundamental programming constructs to develop programs that interact with external components.

Specific Expectations

TF3.01E - define constants, variables, expressions, and assignments statements, including the order in which the operations are performed;

TF3.02E - describe how computers store and work with different types of data, including numbers and characters;

SP3.02E - use a decision structure and a repetition structure in a program.

Planning Notes

- Previous programming structures along with assignment statements and repetition structures are incorporated into problems. Based on the ability level in your class, review the problem sheet (Appendix 4.3.2 – Problem Sheet No. 2) to ensure problems are challenging to all students.
- Review sample teaching examples (Appendix 4.3.1 – Teaching Sample for Repetition) and make any adjustments necessary for language parity.

Prior Knowledge Required

Students require thorough understanding of Activities 1 and 2. Basic knowledge of BEDMAS is helpful.

Teaching/Learning Strategies

- Explain to students the need for repetition structure in programs and the need to identify constants, variables and declare them as comments using good programming style (Appendix 4.3.1 – Teaching Sample for Repetition).
- Review BEDMAS with students and relate this directly to the programming language you are using in class. Explain how equations are used in programming.
- Handout Problem Sheet No. 2 and circulate to assist students with any problems they encounter (Appendix 4.3.2). Students must hand in their best two solutions to these problems with appropriate documentation. It may be appropriate to stop the class at various intervals if several students are having difficulty with the same concept.
- Students complete the assessment question(s) individually, and teachers provide students with written/oral feedback of progress (Appendix 4.3.3 – Formal Assessment).

Assessment/Evaluation Techniques

- Formative assessment using written/oral feedback as students work on problems
- Summative assessment, on two problems students choose to hand in, using the programming rubric (Appendix 4.0 - Programming Rubric)

Accommodations

- Allow students to choose problems to be completed.
- Allow for variable due dates of problem sheets.
- Encourage use of Help menu to assist in problem-solving and extending knowledge.
- Provide enlarged examples of programs for students to follow.
- Revise the formal assessment task as required (Appendix 4.3.3 – Programming Evaluation).

Resources

Print

Carter, John. *Problem Solving in Pascal*. Toronto: Addison-Wesley, 1989. ISBN 0-20111215-9

Elgarten, Gerald, Alfred Posamentors, and Stephens Moresh. *Using Computers in Mathematics*. Toronto: Addison-Wesley, 1983. ISBN 0-201-10450-4

Appendix 4.3.1

Teaching Sample for Variables and Constants

```
REM Variables and Constants
REM Teacher
REM January, 2000
REM Demonstrates variables and constants
DIM Mark, TotalPossible, Percent
CLS
PRINT
INPUT "What is your mark"; Mark
PRINT
INPUT "How many marks were the total possible"; TotalPossible
PRINT
Percent = Mark / TotalPossible * 100
PRINT "Your percent was"; Percent
END
```

Teaching Sample for Repetition

```
REM Repetition Structures
REM Teacher
REM January, 2000
REM Demonstration of Repetition Structures
CLS
PRINT " Simple Loop"
FOR A = 0 TO 5
    PRINT A + 1
NEXT A
CLS
PRINT "Conditional Loops"
N = 1
DO
    PRINT "These are multiples of 10 "; N * 10
    N = N + 1
LOOP UNTIL N > 10
CLS
mark = 50
DO UNTIL mark > 60
PRINT "This mark is rising by the second"; mark
mark = mark + 2
LOOP
END
```

Appendix 4.3.2

Problem Sheet No. 2

Using a problem-solving model and a style guide, write computer programs that:

1. Convert the value of the Canadian dollar against the American dollar. Input an amount of money in Canadian dollars. Output amount in American dollars. The amounts should be formatted in dollars and cents.
2. Use a conversion factor that changes inch measurements into centimetres. Input the length and width of a rectangle in inches. Output the length, width and perimeter in metric form.
3. Input any real value for x . Output the value for y where $y = (x^3 - 3x^2 + 5x)/(x^2)$.
4. Output the natural numbers from 1 to 30. The natural numbers are $\{1, 2, 3, 4 \dots\}$.
5. Input any natural number called number. Output the natural numbers from 1 to number.
6. Input any natural number called number. Output the even natural numbers from 2 to number.
7. Input two natural numbers called number1 and number2 where number2 is bigger than number1. Count down from number2 to number1.
8. Output a chart with the titles NUMBER SQUARE CUBE. Under the appropriate column, output the numbers from 1 to 20, their squares, and their cubes.
9. Recall the problem 3 from Problem Sheet No. 1 and calculate the gross pay for the employees.

Appendix 4.3.3

Problem:

Design and write a program that:

1. Asks a user for his first and last name.
2. Asks a user what their favourite number is between 1 and 9.
3. Prints the times tables from 0 to 12 for their favourite number.

Activity 4: Decision Structure

Time: 420 minutes

Description

Students apply a complete computer problem-solving model to write and document programs that include decision structures. The cumulative programs in this activity use the programming environment from Activity 1, input/output structures of Activity 2, and repetition structures of Activity 3.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes

Overall Expectations

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.03E - explain internal numbering and character representation systems;

TFV.05E - describe the fundamental programming constructs;

SPV.04E - apply fundamental programming constructs to develop programs that interact with external components.

Specific Expectations

TF3.01E - define constants, variables, expressions, and assignments statements, including the order in which the operations are performed;

TF3.02E - describe how computers store and work with different types of data, including numbers and characters;

SP2.03E - trace the operation of a system consisting of a program, an interface, hardware, and directories;

SP3.02E - use a decision structure and a repetition structure in a program;

SP3.03E - design, write, and test a computer program to control a simple peripheral device.

Planning Notes

- Revise appendices to reflect appropriate language and challenge level for your class.

Prior Knowledge Required

Students require a thorough understanding of Activities 1, 2, and 3.

Teaching/Learning Strategies

- Explain the purpose of decision structures in programming. Relate this information to real-world uses in the computer engineering field (e.g., climate controls, traffic lights, timing devices).
- Demonstrate decision statements (Appendix 4.4.1 – Teaching Decision Structures).
- Hand out student problems (Appendix 4.4.2 – Problem Sheet No. 3). Circulate and use questioning techniques to assist students in solving problems with their programs.
- Students hand in their two best programs with appropriate documentation.

Assessment/Evaluation Techniques

- Formative assessment as students work on problem sheet providing written/oral feedback
- Summative evaluation, using the programming rubric, of two problems students hand in (Appendix 4.0 - Programming Rubric)

Accommodations

- Provide enlarged examples of the decision structure programming format.
- Allow students to choose their own problem to solve.
- Exploration into incorporating graphics and sounds in programs may be used.

Resources

Software

Computer programming software (possibly the same software used in Computer and Information Science)

Print

Carter, John. *Problem Solving in Pascal*. Toronto: Addison-Wesley, 1989. ISBN 0-20111215-9

Elgarten, Gerald, Alfred Posamentors, and Stephens Moresch. *Using Computers in Mathematics*. Toronto: Addison-Wesley, 1983. ISBN 0-201-10450-4

Appendix 4.4.1

Teaching Decision Structures

REM Decision Structure

REM Teacher

REM January 2000

REM Demonstrates Decision Structure

CLS

INPUT "How old are you"; Age

IF (Age < 16) THEN

 PRINT "Soon you will be old enough to drive."

ELSE

 PRINT "You are old enough to drive."

ENDIF

END

REM Decision Structure

REM Teacher

REM January 2000

REM Demonstrates Decision Structure

REM IF THEN ELSE

INPUT "Are you having fun yet? (Type Y for Yes or N for No)"; answer

IF (answer = Y) THEN

 PRINT "I am too!"

ELSEIF (answer = N) THEN

 PRINT "I hope this changes for the better!"

ENDIF

END

Appendix 4.4.2

Problem Sheet No. 3

Use a problem-solving model and follow a style guide to write computer programs that:

1. Enter the name of the Capital of Canada. If the answer is Ottawa then display GOOD. If the answer is incorrect then display TRY AGAIN.
2. Ask the user to enter any two numbers. Again ask the user to enter the sum of these two numbers. If the answer for the sum is correct display CORRECT. If the answer is not correct display CHECK THAT SUM.
3. Enter any two words. The two original words should be displayed and then the two words sorted alphabetically.
4. Enter any integer. Display the word NEGATIVE if the number is less than zero, POSITIVE if the number is greater than ZERO, and ZERO if the number is equal to 0.
5. Develop a game relating to guessing the capitals of the provinces in Canada. The conclusion should display the number of correct and the number of incorrect answers. Suitable messages should also be displayed depending on the number of correct answers.
6. Enter a maximum and minimum integer value that could be used in A HIGH LOW GAME. The computer selects a number in the range. The user attempts to guess the number selected by the computer. After each guess the computer displays TOO HIGH or TOO LOW or RIGHT ON depending on the guess. The game ends when the correct number is guessed. The computer then displays the number of attempts to obtain the correct answer and suitable graphic congratulations.

Challengers:

1. Develop a game that allows user interaction. Suitable introductory and concluding credits and graphics as well as announcing the winner should be displayed.
2. Develop a menu screen that displays four options that all relate to displaying patterns using eight circles. The first option allows the user to input an integer from 0 to 255. This Base 10 number is then displayed in binary. The second option counts in binary from 0 to 255. The remaining options are at the discretion of the developer. All four options are to be displayed on screen using eight circles. Draw eight evenly spaced non-overlapping circles that extend across the middle of the screen. The eight circles represent the eight bits in one byte.

Unit 5: Computer Interfacing

Time: 30 hours

Unit Description

The final culminating unit incorporates information learned in all previous units. Students are expected to work through Activities 1, 2, and 3 to fully prepare them for the challenge of Activity 4 where they complete a project that demonstrates their knowledge of integrating software and hardware processes to solve an interfacing challenge.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Processes, Impact and Consequences

Overall Expectations: TFV.01E, TFV.02E, TFV.03E, TFV.04E, TFV.05E, SPV.01E, SPV.02E, SPV.03E, SPV.04E, ICV.02E.

Specific Expectations: TF1.01E, TF1.02E, TF1.03E, TF2.01, TF2.02E, TF2.03E, TF2.04E, SP1.01E, SP1.02E, SP2.02E, SP2.03E, SP3.01E, SP3.02E, SP3.03E, IC1.02E, IC1.09E, IC1.10E.

Activity Titles (Time + Sequence)

Activity 1	What is Interfacing?	180 minutes
Activity 2	Interfacing Software	300 minutes
Activity 3	Interfacing Hardware Design	300 minutes
Activity 4	Designing, Building, and Operating an Interface System	1020 minutes

Prior Knowledge Required

Students have computer hardware experience from Unit 1, network hardware and software experience from Unit 2, hands-on integrated circuit experience from Unit 3, and programming experience from Unit 4. By this time in the course, it is also expected that students have achieved a comfort level to be able to work independently when required.

Unit Planning Notes

The focus of this unit is applying the programming software skills from Unit 4 to hardware and interfaces. Activities 1, 2, and 3 give students the knowledge and skills they require to complete Activity 4 which is designing, building, and operating an interface system. It is helpful to have completed projects for students to see and use, and magazines or resources for them to use as reference materials. It is important to stress that simple interface designs are required in this introductory computer engineering course and to encourage students to avoid elaborate designs. The software and computer programming languages used depend upon the software available in the school but must be capable of controlling the mouse and parallel port. The types of electronic components available to build the hardware must be considered as well as ethical issues that can result. A field trip to visit a manufacturer or other business that employs computerized machinery would be beneficial. Inviting a guest speaker to your class would also help to educate students to the range of possibilities of interfacing computer systems. You may choose the final activity for all students to create however, there are possibilities for students to generate and create their own ideas. Although a PC is the type of computer used, consider ideas using micro-controllers and PLCs as they are used extensively in industry to interface with many external systems and devices. The final activity could be the final evaluation for the course, combined with the student's Portfolio and glossary of terms.

Teaching/Learning Strategies

The delivery of this unit requires good planning and structure for students who are not able to work independently on a project, yet allow room for more independent students to work through the design process from beginning to end with the teacher facilitating. This unit includes conferencing, peer practice and teaching, research, class discussions, process notes, hands-on computer interfacing activities, and Socratic lessons.

Assessment and Evaluation

Assessment and evaluation are based on hands-on activities. The Learning Skills Checklist found in Unit 1 should be used throughout this unit. Formative assessment is ongoing as students work toward a final project. The final evaluation is based on the rubric in Appendix 5.0. Students should also use this rubric to assess their progress and final product. An individual performance assessment is included in the final activity as well.

Resources

Software

Computer programming software (possibly the same software used in Computer and Information Science)

Word-processing and database software

Print

Baker, J. *Digital Computer Technology: An Introduction*. Toronto: McGraw-Hill Ryerson, 1983. ISBN 0-13-211947-1

Barbarelo, James. *Real World Interfacing with Your PC*. Indianapolis: Howard W. Sans & Co., 1996. ISBN 0-7906-1145-7

Bergsma, Paul. *Controlling the World with Your PC*. California: High Text Publications Inc., 1994. ISBN 1-878707-15-9

Lawrence, Orville. *Computer Technology*. Toronto: McGraw-Hill Ryerson, 1984. ISBN 0-07-548711-X

Powers, Thomas. *The Integrated Circuit Hobbyist's Handbook*. California: High Text Publications, Inc., 1995. ISBN 1-878707-12-4

Shelly, Gary and Thomas Cashman. *Computer Fundamentals for an Information Age*. California: Anaheim Publishing Co., 1984. ISBN 0-88236-125-2

Simms, Forrest. *Getting Started in Electronics*. USA: Radio Shack, 1983. Cat No 276-5003

Tokheim, Roger. *Digital Electronics*. Toronto: McGraw-Hill Book Company, 1984. ISBN 0-07-064980-4

Web Sites

Parallel port Central

<http://www.lvr.com/parport.htm>

Appendix 5.0

Activity Rubric

Criteria	Level 1 (50-59%)	Level 2 (60-69%)	Level 3 (70-79%)	Level 4 (80-100%)
Knowledge/Understanding				
Understanding of concepts	- demonstrates limited understanding of computer component concepts	- demonstrates some understanding of computer component concepts	- demonstrates considerable understanding of computer component concepts	- demonstrates thorough and insightful understanding of computer component concepts
Thinking/Inquiry				
Inquiry/design skills	- applies few of the skills needed to solve problems	- applies some of the skills needed to solve problems	- applies most of the skills needed to solve problems	- applies all or most of the skills required to solve problems
Follows design process	- beginning stages of design process apparent	- follows most stages of the design process	- follows all stages of the design process	- follows all stages of the design process including contingency plans
Communication				
Written documentation	- communicates written information with limited clarity	- communicates written information with moderate clarity	- communicates written information with considerable clarity	- communicates written information with a high degree of clarity and confidence
Complete diagrams (including labels)	- some diagrams are included and are complete	- most diagrams are included and are complete	- all diagrams are included and are complete	- all diagrams are included, complete and are of high quality
Application				
making connections in input, processing, and output	- makes connections between computer hardware, software, and processing with limited effectiveness	- makes connections between computer hardware, software, and processing with moderate effectiveness	- makes connections between computer hardware, software, and processing with considerable effectiveness	- makes connections between computer hardware, software, and processing with high degree of effectiveness
incorporates ideas presented throughout the course	- incorporates few ideas presented in the course	- incorporates several ideas presented in the course	- incorporates ideas presented in the course effectively	- incorporates ideas presented in the course in an innovative way

Note: A student whose achievement is below level 1 (50%) has not met the expectations for this assignment or activity.

Activity 1: What is Interfacing?

Time: 180 minutes

Description

Students explore hardware and software considerations in designing an interfacing system. They use a word-processing and/or database application to research common available electronic components used in interfacing. Students also begin a list of possible projects they could build during the hands-on construction of an interface system in Activity 4.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Impact and Consequences

Overall Expectations

TFV.01E - describe how the internal components of the computer enable the peripherals to function;

TFV.02E - describe a problem-solving model such as the input, processing, output model;

ICV.02E - identify the social impact of computers and associated technologies.

Specific Expectations

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;

TF2.02E - identify the basic internal and external components of a computer;

TF2.03E - describe the primary function of each component;

TF2.04E - identify computer internals and peripheral devices and describe their relationships;

IC1.09E - demonstrate compliance with acceptable-use policies.

Planning Notes

- Review small group presentation ideas to include any you may have to share with the students. You may also wish to change the names to those of students in your class (Appendix 5.1.1 – Small Group Interfaces Presentation Ideas).
- Have available interfacing components available for students to see and explore (e.g., connectors, cables, LEDs, resistors).
- Source local (if available) interface and electronic component manufacturers for brochures, videos, and presentations of computerized processes that demonstrate interfaces to support classroom theory.

Prior Knowledge Required

- Hardware activities from Unit 1
- Networking terminology from Unit 2
- Hands-on integrated circuit activities from Unit 3

Teaching/Learning Strategies

- Define interface for students (interface can be hardware or software that interconnects components of a computer system (i.e., PIC, PLC, or PC) or connects two computers to each other). Using the definition and in groups of two or three, students decide whether or not the statements in Appendix 5.1.1 are demonstrating an interface or not. They must provide reasons for their choice (Appendix 5.1.1 – Small Group Interface Presentation Ideas).
- Students need to understand that the concept of interconnectedness is not a one-way process. In terms of a printer connection, messages come back to the computer if there are problems with the printing process. In this case the peripheral is now the input device.

-
- Student teams of two to four explore what could be hooked up to a computer using a cable. Students experiment to see which connections can be made using a process of input (component gives instructions), processing (how the instruction is moved from one place to another) and output (component receives and interprets instruction).
 - From available materials, students build an interface connection using the above formula (e.g., using a cable to connect a computer and a printer). It may also include micro-controllers, PLCs, as well as circuits they have built in previous units.
 - Students are then directed to resource materials (catalogues, brochures, manuals, Internet sites) to list components used in interface designs. They fill in the chart to gain more knowledge of materials available (Appendix 5.1.2 – Interface Components). Students add to their glossary of terms.
 - Finally, students begin the design process for their project in Activity 4. Students are reminded of the design process (Appendix 5.1.3 – Design Brief). They are not making choices, however, they are gathering information to make informed design decisions. This planning process should be kept in their portfolio.

Assessment/Evaluation Techniques

- Learning Skills Checklist
- Checklist assessment of building an interface (Appendix 1.1.7 – Activity Checklist may be adapted).
- Formative assessment giving students written/oral feedback on the beginning thoughts in their design process to give them pointers toward success.

Accommodations

- Pre-teach vocabulary and definitions prior to and in context of lesson when necessary.
- Monitor individual submissions and provide individual feedback.
- Provide word lists and definitions when necessary.

Resources

Software

Ministry licensed word processor or spreadsheet

Print

Barbarelo, James. *Real World Interfacing with Your PC*. Indianapolis: Howard W. Sans & Co., 1996. ISBN 0-7906-1145-7

Bergsma, Paul. *Controlling the World with Your PC*. California: High Text Publications Inc., 1994. ISBN 1-878707-15-9

Simms, Forrest. *Getting Started in Electronics*. USA: Radio Shack, 1983. Cat No 276-5003

Web Sites

Parallel port Central

<http://www.lvr.com/parport.htm>

Appendix 5.1.1

Small Group Interface Presentation Ideas

Student groups:

- a) state whether or not this is an interface and if yes what the interface is;
 - b) give reasons for their choice.
1. Student 1 wishes to have a copy of the group's project that is saved on Student 2's computer. He uses a disk and downloads the information and then copies it to his computer (not an interface – not interconnected).
 2. Student 1 sends via e-mail the group's project for Student 2 to save on his system and then in turn posts it on the web site monitored by their group.
 3. Jane purchased a new keyboard for her computer, however, the connector did not match the port on her computer. Looking further Jane found a second connector in the box that converted her old system to accept the new keyboard connection.
 4. Bill has a special project at work that requires he use information in his employee database, transfer it to a spreadsheet to complete necessary calculations, and present that information globally on the company's web site. His computer department has created a program that will connect the information he requires by a few clicks of his mouse.
 5. Jennifer works for a large pharmaceutical company. Every now and then, Jennifer must change the assembly line set up for a new product they are manufacturing. Because all the equipment is computerized, it means that new programs must be written or old programs changed to accommodate the new product.
 6. Maury is printing a huge complex order at his printing firm. In order to print during the night and not have to worry about printer problems, he has connected more than one printer to his system. He wrote a simple program that instructs the computer to switch printers if one should jam, run out of paper, or run out of ink.
 7. Dhayalan and Shiori are working on a project in their computer engineering course. Dhayalan is using *Electronic Workbench* to test their circuit and create the schematics. Shiori is documenting their work using a word processor. They present their project together (not an interface – not interconnected).
 8. New traffic lights were installed outside the public school on Edward Street to assist students crossing. Rhonda, who is the traffic engineer with the public works department, must be sure that the new lights are sequenced with the lights at the corner and do not come on during the weekend when no students are using the crosswalk.

Appendix 5.1.2

Interface Components

Component	Manufacturer	Software/Hardware	Industrial Application

Appendix 5.1.3

Design Brief

The SPICE Model

S – Situation

P – Problem

I – Investigation

C – Construction

E – Evaluation

Situation (Scenario)

The situation gives the frame of reference for the design brief or problem. This sets the stage for the activity and helps to provide a context for the design problem. The situation should be a creative story that takes into account such factors as age and background knowledge of the students. Students are encouraged to write situations that are as real as possible while maintaining an element of creativity. The situations should be written so they are open-ended and allow for a variety of solutions.

Problem (Design Brief)

The design brief is a short statement indicating what is to be designed and made. A detailed design brief also states any restrictions and/or requirements placed on the design such as materials, size, cost, time allotted, and processes to be used. The design brief is developed in conjunction with the situation.

Investigation

During this stage, students incorporate a variety of idea generation and information gathering techniques to learn more about the problem and a possible solution. Students brainstorm a variety of possible solutions and consult reference materials for further ideas. Students should sketch their ideas for solutions and include annotations as needed to help explain how the solution works and is fabricated. In conducting their investigation, students consider such factors as cost, size, availability of materials, appearance, functionality, ergonomics, and safety considerations while ensuring that all requirements of the design brief are met.

Construction

Once students have compiled a list of ideas they will select the one they feel is the best possible solution to the problem. Students then complete working diagrams of their solution with all the necessary dimensions. They choose the best possible method of construction and then begin fabricating their solution. Students should first build a prototype from recyclable materials in order to ensure their solution is viable.

Evaluation

During this stage, students develop methods of testing their prototype to see if it meets the design criteria. If it does then they begin the construction of the final product. If it fails to meet the requirements then students should return to the investigation and choose another solution or modify the existing one. The new solution must be constructed and tested again to ensure the design brief requirements have been satisfied.

Activity 2: Interfacing Software

Time: 300 minutes

Description

Students use prior knowledge combined with new information to comprehend how communication occurs between a computer and a peripheral device and the role of the interface in this process. Students design and write programs to accomplish specific simple tasks with the peripheral device.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Process

Overall Expectations

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.03E - explain internal numbering and character representation systems;

TFV.05E - describe the fundamental programming constructs;

SPV.02E - demonstrate the use of an operating system, including a network;

SPV.04E - apply fundamental programming constructs to develop programs that interact with external components.

Specific Expectations

TF1.01E - describe the relationship between the binary number system and computer logic;

TF1.02E - define a standard way of representing characters in a binary code;

TF1.03E - describe the function of the fundamental logic gates, including the function of each pin: AND, NAND, OR, NOR, XOR, XNOR, and NOT;

SP1.01E - convert positive integer numbers to binary form;

SP1.02E - derive the truth tables of the fundamental logic gates;

SP3.01E - use input and output statements in a program;

SP3.02E - use a decision structure and a repetition structure in a program;

SP3.03E - design, write, and test a computer program to control a simple peripheral device.

Planning Notes

- Select suitable software that supports programming graphics as well as provides access to the parallel port and mouse (e.g., Turing, Pascal, Basic, C). Note that using the mouse enhances the activity for the students, however, this may be done with keyboard input as well.
- Review graphics capabilities of software.
- Consideration must be given to possible conflicts between the operating system software and programming software as to which has control of the parallel port.
- Consideration must also be given to possible ethical and moral issues when students have access to the operating system while programming.

Prior Knowledge Required

- Unit 3, hands-on integrated circuit activities
- Unit 4, computer programming activities

Teaching/Learning Strategies

- Begin this activity with a question aimed at assessing student knowledge on how computerized systems are used in manufacturing or industrial work places. (Who has ever seen a computerized process (robotic arm, etc.) at work?) The emphasis is to discover the knowledge base and the depth of knowledge.
- Students are to provide information on the function of the parallel port and functions of the pins in order to understand how communication is interpreted through the interface to the peripheral device (Appendix 5.2.1 – Interface Commands). Students need to understand how programming the pins in the parallel port affect what is happening to or from the peripheral device.
- Students complete a review of binary operations and schematic drawings using paper and pencil activities.
- Teachers review with students some of the repetition structures and decision-making structures from Unit 4. Students, in small groups, use an input, processing, output model to give a possible scenario of what types of instructions a computer might send to a peripheral and what instructions the peripheral might send back. Students share their thoughts with the class and discuss the merits and possible problems with each scenario.
- Students write programs that send a message and have capability of responding to input from a peripheral (e.g., the computer signals the peripheral device to count items and when a certain number is obtained a message appears and the counting process ends). Students build simple circuits (or use previously designed circuits), attach them via a parallel port, and program signals to their new peripheral device. If using mouse control refer to (Appendix 5.2.2 – Mouse Control).
- Finally, students add to their design brief ideas that they could use to program a peripheral device. They begin to assess the materials available for the construction stage. Again, remind students they are not making decisions but gathering information and creating ideas for their final project.

Assessment/Evaluation Techniques

- Formative assessment on student programs (Rubric 4.0 – Programming Rubric)
- Formative assessment of design ideas kept in student Portfolio. Use questioning techniques to help them accept or reject choices for the final project.
- Formative assessment on paper and pencil tasks indicating clearly to students their strengths and possible areas for improvement before they start their final project.

Accommodations

- Students with programming experience from Grade 9 Integrated Technologies could assist other students or begin constructing interfaces.
- Provide specific programming template where necessary.
- Adapt time constraints on exercises where necessary.
- Have programs available for students to adapt or modify.
- Provide visuals of industrial use PLCs, PICs, PCs, and robotics.

Resources

Software

Computer programming software (possibly the same software used in Computer and Information Science)

Print

Barbarelo, James. *Real World Interfacing with Your PC*. Indianapolis: Howard W. Sans & Co., 1996. ISBN 0-7906-1145-7

Bergsma, Paul. *Controlling the World with Your PC*. California: High Text Publications Inc., 1994. ISBN 1-878707-15-9

Tokheim, Roger. *Digital Electronics*. Toronto: McGraw-Hill Book Company, 1984. ISBN 0-07-064980-4

Web Sites

Parallel port Central

<http://www.lvr.com/parport.htm>

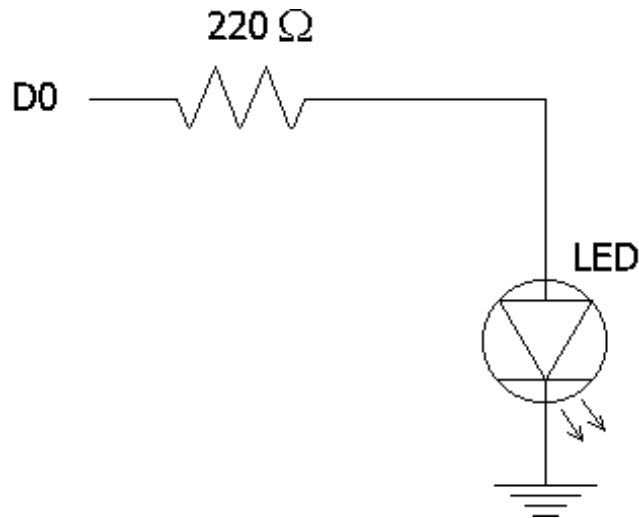
Appendix 5.2.1

Interface Commands

An interfacing system consists of a computer, interface, and peripheral. The computer is able to communicate with the interface through the parallel port. The computer sends commands (signals) to the parallel port through software. The software can be written in any of several languages (such as Basic, Pascal, C, and Turing). The language chosen should easily communicate with the parallel port. (It should also support graphics and access the mouse commands if you are using a mouse control.) Pin configurations were introduced in Unit 2.

1. How many pins on the parallel port are normally reserved for outputting data?
2. What is the smallest number that can be output on the parallel port?
3. What is the largest number that can be output on the parallel port?
4. When a Base 10 number is sent to the parallel port, the binary equivalent is actually outputted on pins labelled D0, D1, D2 ... D7. Complete the chart that relates the Base 10 number and the equivalent number on the output pins of the parallel port.

Number	D0	D1	D2	D3	D4	D5	D6	D7
1								
15								
255								
18								



Appendix 5.2.2

Mouse Control

Mouse and graphic commands are interrelated. The graphic commands are useful to display diagrams that represent the actual real-world peripheral being controlled by the computer. The mouse commands are also useful to select graphic options displayed on the screen. When an option is selected by clicking, the software takes appropriate action and issues commands through the parallel port. The real-world object being controlled is activated at the same time as the screen graphic. Graphical User Interfaces (GUIs) are very popular commercially.

Question 1

Write a program to display one LED on the screen.

Question 2

Write a program to represent the one LED turning on for three seconds and then turning off.

Question 3

Write a program that represents the one LED turning on for three seconds and then turning off for two seconds. This process should repeat twenty times.

Question 4

Write a program that displays three small boxes and an LED on the screen. The first box contains the word ON. The second box contains the word OFF. The third box contains the word EXIT. The program should turn the LED on when the first box is clicked, off when the second box is clicked, and exit the program when the third box is clicked.

Question 5

Write a program that graphically displays eight LEDs (one byte). Input any integer from 0 to 255. Display the binary equivalent on the LEDs.

Question 6

Write a program that graphically displays eight LEDs (one byte). Develop a program that counts from 0 to 255 on the LEDs.

Activity 3: Interfacing Hardware Design

Time: 300 minutes

Description

Students design an interface and a related peripheral. They learn the function of the interface is to translate signals between the computer's parallel port and the peripheral. The interfaces designed in this activity are incorporated into hands-on building of an interfacing system in Activity 4.

Strand(s) and Expectations

Strand(s): Theory and Foundation

Overall Expectations

TFV.01E - describe how the internal components of the computer enable the peripherals to function;

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.04E - describe and illustrate the functions of logic gates.

Specific Expectations

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;

TF2.02E - identify the basic internal and external components of a computer;

TF2.03E - describe the basic internal and external components of a computer;

TF2.04E - identify computer internals and peripheral devices and describe their relationships.

Planning Notes

- Select suitable methods to design interfacing hardware (e.g., pencil/paper, computer-assisted design).
- Design must consider the electronic components available.
- Consider possible projects for Activity 4.
- Consider interfacing to micro-controller circuits especially if you have used them in previous units.

Prior Knowledge Required

- Computer hardware components from Unit 1
- Hands-on integrated circuit activities from Unit 3
- Availability of electronic components discussed in Unit 5, Activity 1

Teaching/Learning Strategies

- The final stage before students begin their final project is for students to trace how a system operates using a program, interface, hardware, and directories. Using prior student knowledge, reconstruct how a stand-alone computer system functions using all components (interface can be between two software programs).
- Using simple set-ups connecting computers to motors where programs are already available, students work through the details, tracing the system using correct terminology and electronic descriptions. (Note: It is acceptable for students to use pictures rather than writing.)
- Describe different applications of computerized processes used in industry today. Teach students to break down these ideas into a project they can manage. For example: In a warehousing situation, a robotic arm may pick up boxes and position them on a conveyor belt. Each time it picks up a box, it sends a signal to the computer controller to add one to the count. As the process moves on, the computer stops the arm when it has reached a certain level. This may then produce a signal that a new task is ready to start. To change this into student terms, they could have a switch as a peripheral. Their program counts the number of times the switch is turned on and off and, upon reaching a certain number (e.g., five), the program displays a message (e.g., Limit has been reached!).

-
- Students choose one problem to follow the full design process. They must design and draw a simple schematic diagram, describe the software routine, and source the components required using available research tools (magazines, web sources, catalogues). Encourage students to use the easiest and most cost effective method to solve their problem. If time permits and resources are readily available, students may build and test their choices. You may wish to use the interfacing one bit activity (Appendix 5.3.1 – Interfacing One Bit). Appendix 5.3.2 is specific to micro-controllers, however, it could be easily adapted for use with a PC (Appendix 5.3.2 – Interfacing to Micro-controller Circuits).
 - Through set-ups that the students have seen (a field trip to computerized environments would be ideal), they now complete a plan for their unit four project. Students have been planning throughout this unit. Although the teacher may assign a specific task for the final project, students' previous plans assist in personalizing the task. This project is done individually, however students may use teacher and peer input to strengthen their ideas.

Assessment/Evaluation Techniques

- Formative evaluation on student interface designs (Appendix 5.0 – Activity Rubric). Ensure students clearly understand their strengths, areas needed for improvement, and steps they need to include to achieve success in the final project.
- Learning Skills Checklist

Accommodations

- Provide step-by-step or visual instructions.
- Provide examples of designs, schematics, and proper documentation.
- Use exemplars of previous student work.
- Choose different interface activities for students at different levels of achievement, (e.g., simple on-off motor control for some, variable speed motor control for others, and variable speed, reversing motor control for exceptional students).

Resources

Software

Computer programming software (possibly the same software used in Computer and Information Science)

Print

Barbarelo, James. *Real World Interfacing with Your PC*. Indianapolis: Howard W. Sans & Co., 1996. ISBN 0-7906-1145-7

Bergsma, Paul. *Controlling the World with Your PC*. California: High Text Publications Inc., 1994. ISBN 1-878707-15-9

Powers, Thomas. *The Integrated Circuit Hobbyist's Handbook*. California: High Text Publications, Inc., 1995. ISBN 1-878707-12-4

Tokheim, Roger. *Digital Electronics*. Toronto: McGraw-Hill Book Company, 1984. ISBN 0-07-064980-4

Electronics Trade Magazines:

PopTronics

<http://www.gernsback.com>

Circuit Cellar Ink, The Computer Applications Journal

<http://www.circuitcellar.com>

Web Sites

Parallel port Central – parallel port questions, answers, links, and activities

<http://www.lvr.com/parport.htm>

Micro-controller manufacturer web sites:

Atmel Corporation home page

<http://www.atmel.com>

Microchip Technology Inc. home page

<http://www.microchip.com>

Motorola Semiconductor Products Group home page

<http://www.mot-sps.com/products/index.html>

Philips Semiconductors home page

<http://www-us.semiconductors.philips.com/mcu/products>

Scenix Semiconductor Inc. home page

<http://www.scenix.com>

Zilog Inc. home page

<http://www.zilog.com>

Internet micro-controller resources:

Webring directory computer hardware page

<http://www.webring.org/ringworld/comp/hardware.html>

eg3.com embedded systems directory page

<http://www.eg3.com>

PIC micro programmers and development tools:

Microchip Technology Inc. home page

<http://www.microchip.com>

Microchip's MPLAB is a free PICmicro cross-assembler and simulator.

microEngineering Labs, Inc. home page

<http://www.microengineeringlabs.com/mel/home.htm>

microEngineering Labs is a manufacturer of simple, low cost, PICmicro programmers and development boards.

Parallax Inc. home page

<http://www.parallaxinc.com>

Parallax manufactures the BASIC Stamp, Scenix programmers and educational micro-controller projects.

Tech Tools home page

<http://www.tech-tools.com>

Tech Tools manufactures a number of embedded control programmers and simulators.

Digi-Key home page

<http://www.digikey.com>

Digi-Key distributes PICmicro micro-controllers and programmers.

Schematic entry and printed circuit board layout software vendors:

ORCAD home page

<http://www.orcad.com>

ORCAD has a free, limited, version of their integrated design software.

Electronics Workbench home page

<http://www.electronicworkbench.com>

Press-n-Peel iron-on printed circuit board transfer film.

Techniks Inc. home page

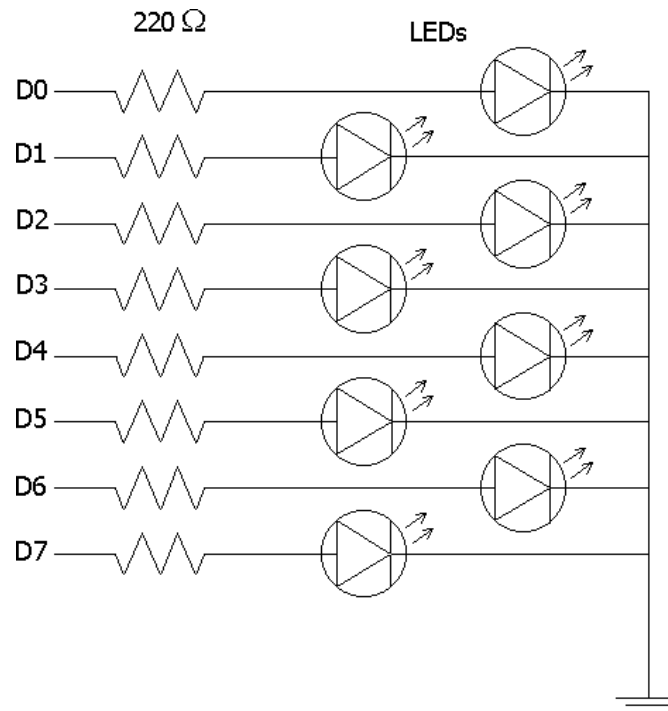
<http://www.techniks.com>

Use a laser printer with Techniks Press-n-Peel Blue for iron-on circuit patterns.

Appendix 5.3.1

Interfacing One Bit

Analyse the design of an interfacing system consisting of one LED controlled by output pin D0 of the parallel port.



- 1) What does D0 represent?
- 2) How is the computer connected to the resistor?
- 3) What would the colour bands on the resistor be?
- 4) How is the resistor connected to the LED?
- 5) Which lead on the LED connects it to the resistor?
- 6) Which lead on the LED is connected to ground?
- 7) An interfacing system consists of three main components: computer, interface, and peripheral.
 - a) What comprises the interface in this example?
 - b) What comprises the peripheral?

Appendix 5.3.2

Interfacing to Micro-controller Circuits

Description

In this activity the students design and build an interface circuit, connect it to a micro-controller experiment board (designed and built as part of the Integrated Circuits unit), and program the interface using the assembly code skills developed in the Programming unit. During the completion of this unit, students must recall hardware details, circuit operation, and programming techniques, allowing these unit activities to form an ideal final project that integrates all of the components of Computer Engineering Technology.

Teaching/Learning Strategies

- The computer interfacing unit is by its nature very student-centred and open to experimentation. Once students understand the flexibility and power of a micro-controller, they can easily become interested in applications that use micro-controllers. However, the difficulty of programming in assembly code necessitates a very structured and orderly approach to software and hardware design. Students should be encouraged to start with simple interface circuits, or to break a complex project into smaller simpler parts, so that it becomes easier for them to slowly build their programming expertise and to experience a number of successes along the way.
 - Due to the inherent flexibility afforded by a great variety of available electronic and interface components, this activity can only provide general suggestions for planning and implementing interface circuits. The skill and competency of the students, for the most part, determines the actual type and complexity of the interface circuits to be built. Simple projects can be done by individual students, more complex projects are best tackled in teams.
1. As a first step, the teacher should describe, and insist that the students follow, a prescriptive design process. The primary purpose of the design process is to create a general description of the project solution. After completing the design process, the students should have a better grasp of how attainable their design goals might be.
 2. After completing the general design process, students create a specific functional description of their project, its operation, and its man-machine interface. At this point, natural language descriptions are preferred for their ease of comprehension. Get students to list and describe what the project is designed to do, and how they think the micro-controller can accomplish this.
 - a) For example, a student project might be variable speed control of a fan motor. The student would draw a simple schematic diagram to describe how to connect a high current transistor drive circuit to a micro-controller output. Then the student would describe the software routine that pulse modulates the transistor to make it control the motor speed. The student also needs to describe what type of input signal is used to determine the motor speed (e.g., an electronic thermostat controls the motor's speed).
 - b) The teacher can help lead or direct students toward circuits and circuit ideas. In the variable speed motor example, above, help the students examine the motor circuits in old CD players, VCRs, and power tools.
 3. Next, students use available data research tools to select components used to construct the interface. The students should be encouraged to find the simplest solution that most effectively solves their problem. Monitoring project costs and product delivery times becomes important at this point. If the components are too expensive, or not readily available, the project can still be easily changed because of the relatively low amount of time and energy spent on these steps so far. Once specific components have been selected, students should begin drawing a detailed design and keeping a record of all parts and costs.

Appendix 5.3.2 (Continued)

Interfacing to Micro-controller Circuits

4. If the interface parts are available, have the students design, build, and test a circuit, much as they did when wiring gate circuits or building the micro-controller experiment board in the Integrated Circuits unit. Students should determine how to best evaluate the circuit to ensure that it effectively solves their design problem. Finally, the operation of the test circuit should be verified, and detailed measurements should be taken to assist in the creation and debugging of the actual interface project.
5. While the interface hardware is being constructed, have the students develop a detailed software flowchart, specifically detailing the algorithms necessary to program the micro-controller. Students need to use micro-controller instructions to match the natural language functional description of their interface developed in step 2.
6. With the interface hardware and software flowchart complete, the students can begin the task of programming, testing, and debugging. The programming unit introduced students to a systematic method of software design. By following these same steps, they complete the program for their interface project.

Activity 4: Designing, Building, and Operating Interface Systems

Time: 1020 minutes

Description

Students integrate the software and hardware theory and practice from previous units to construct a simple two-wheel robot mouse that is controlled by a PC parallel port. Students are able to test their final product during a keyboard controlled robot race through a mouse cheese maze.

Strand(s) and Expectations

Strand(s): Theory and Foundation, Skills and Process, Impact and Consequences

Overall Expectations

TFV.01E - describe how the internal components of the computer enable the peripherals to function;

TFV.02E - describe a problem-solving model such as the input, processing, output model;

TFV.04E - describe and illustrate the functions of logic gates;

SPV.01E - connect and use correctly a variety of computer components and peripherals;

SPV.03E - use logic gates to construct simple circuits.

Specific Expectations

TF1.03E - describe the function of the fundamental logic gates, including the function of each pin: AND, NAND, OR, NOR, XOR, XNOR, and NOT;

TF2.01E - use precise terminology in relation to all hardware, interfaces, and networking systems;

TF2.02E - identify the basic internal and external components of a computer;

TF2.03E - describe the basic internal and external components of a computer;

TF2.04E - identify computer internals and peripheral devices and describe their relationships;

SP1.02E - derive the truth tables of the fundamental logic gates;

SP2.02E - build an interface to connect the computer to a simple peripheral device;

SP2.03E - trace the operation of a system consisting of a program, an interface, hardware, and directories;

SP3.03E - design, write, and test a computer program to control a simple peripheral device;

IC1.02E - use safe practices in the handling of computer hardware and electronic components.

Prior Knowledge Required

- Unit 1, Hardware
- Unit 2, Networking
- Unit 3, Integrated Circuits
- Unit 4, Programming
- Unit 5, Activities 1, 2, and 3

Planning Notes

- Teachers should be aware that some students may find this activity very challenging.
- Develop other suitable interfacing projects considering hardware and software limitations.
- Consider ethical issues which may emerge such as stolen parts or projects.
- Gather pricing information of electronic components to assemble interfaces and peripherals.
- A test program running on a computer test station is set up by the teacher for students to verify the Programmable PC Parallel Port Output Interface as functioning correctly.
- Students may do planning and designing as an individual task, but it is recommended that students work in teams of two to four for the duration of this activity.

Teaching/Learning Strategies

- Teachers must ensure all safety passports are up to date and include all safety lessons.
- Teachers instruct students, that the portfolio and glossary of terms are evaluated at the end of this unit along with their final project. Students may finalize these two items if they have any down time during the activity otherwise they must complete for homework.
- Introduce the MICE ROBOT CHEESE MAZE challenge.
- Form teams to build team spirit with teams choosing a name and a motto. Groups should consist of students assigned specific tasks to named job titles. Job titles and associated tasks are defined for the class.

Project Manager – overall team leader with specific responsibility for scheduling and project tracking

Chief Programmer – C control code

Electrical Engineer – computer interface wiring and motor control circuitry

Draftsperson – drawings and project documentation

Test Technologist – hardware and software test and evaluation

Assembly Technician – building Mouse Robot and interface

Mouse Maze Racer – runs the completed robot.

- Once teams have been established, each student is given multiple job assignments. The primary purpose of the named job titles and associated tasks is to guide students to a successful project management structure. The teacher must facilitate team tasking and monitor each student's activity and overall group progress on an ongoing basis.
- Develop an activity schedule with the class and hand out a personal daily activity reporting log that must be turned in daily. Indicate specific key dates logically scheduled for preliminary drawings, documentation, programs, and hardware to ensure successful completion of the robot challenge (Appendix 5.4.1 – Student Activity Reporting Log).
- Provide a sample of an interface hardware and design details.
- Provide each team with an electromechanical hardware kit to design and manufacture their robot. Students may create their own kits from a materials list given them by the teacher.
- Teachers set up a sample or test maze (e.g., a taped pattern on a floor tile grid) and encourage one-on-one robot races or pushing/shoving challenges with prototype electromechanical design.
- Keep everybody busy and have fun!

Assessment/Evaluation Techniques

- Formative assessment as students work to give them feedback on progress and suggest areas needed for improvement for their success
- Summative evaluation of student portfolio and glossary of terms
- Summative evaluation on the design, implementation, and functionality of a student interfacing system (Appendix 5.0 – Activity Rubric)

Accommodations

- Students with suitable programming experience could develop additional GUI options.
- Have a complete system in place for students to observe.
- Make programming task as simple as editing existing code to include the team name, members and job titles.
- Provide pre-built and tested parallel port output interface control card.
- Provide extended race practice time with mouse robot.
- Provide surplus printed wiring boards for solder and desoldering skill building activity.
- Build necessary interface cables.

-
- Students could program a two-wheeled robot to run autonomously through a stored movement sequence.
 - Students could modify the program to change the up/down/right/left arrow keys to the number pad to provide a gentler turn on the 1/3/7/9 keys and forward/back/right/left on the 8/2/6/4 keys.
 - Where facilities permit have the student manufacture their printed wiring board from a proven copper foil mask.
 - Add computer-controlled lights and horn to the robot.
 - Troubleshooting other student interface boards can be a rewarding bonus activity.

Resources

Software

Computer programming software (possibly the same software used in Computer and Information Science)

Print

Baker, J. *Digital Computer Technology: An Introduction*. Toronto: McGraw-Hill Ryerson, 1983. ISBN 0-13-211947-1

Barbarelo, James. *Real World Interfacing with Your PC*. Indianapolis: Howard W. Sans & Co., 1996. ISBN 0-7906-1145-7

Bergsma, Paul. *Controlling the World with Your PC*. California: High Text Publications Inc., 1994. ISBN 1-878707-15-9

Lawrence, Orville. *Computer Technology*. Toronto: McGraw-Hill Ryerson, 1984. ISBN 0-07-548711-X

Powers, Thomas. *The Integrated Circuit Hobbyist's Handbook*. California: High Text Publications, Inc., 1995. ISBN 1-878707-12-4

Shelly, Gary and Thomas Cashman. *Computer Fundamentals for an Information Age*. California: Anaheim Publishing Co., 1984. ISBN 0-88236-125-2

Simms, Forrest. *Getting Started in Electronics*. USA: Radio Shack, 1983. Cat No 276-5003

Tokheim, Roger. *Digital Electronics*. Toronto: McGraw-Hill Book Company, 1984. ISBN 0-07-064980-4

Web Sites

Parallel port Central

<http://www.lvr.com/parport.htm>

C Programming Training – introductory I/O C programming training

<http://www.madnetter.com>

Circuit Cards – Simple parallel port output interface circuit card

<http://www.siriusmicro.com>

Appendix 5.4.1

Student Activity Reporting Log

Teachers provide students Day # and specific dates for preliminary drawings, documentation, programs, and hardware construction to ensure successful completion of activity.

Team Name:			Start Date:	
Motto:			Project Completion Date:	
Members:				
Day # and Activity	Contributions by:	Problems encountered and solutions	Due Date	Date Completed
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14 Race Day				