

# **Introduction to Engineering & Computer Science**

*“with emphasis on Computing and Electrical Fields”*

*Version 4.0 printed on September 2012  
First published in September 2006*

## **Background and Acknowledgements**

This text is prepared for students who intend to start their education in Engineering or Computer Science fields with emphasis in Electrical Engineering, Computer Engineering and Computer Science. The content is derived from the author's educational, technical and management experiences, in-addition to teaching experience. Many other sources, including the following specific sources, have also informed the content and format of this text:

- Katz, R. Contemporary Logic Design. (2005) Pearson.
- Lumsdaine, E. Creative Problem Solving and Engineering. (1999) Prentice Hall.
- Wakerly, I. Digital Design. (2001) Prentice Hall.
- Sandige, R. Digital Design Essentials. (2002) Prentice Hall.
- Nilsson, J. Electrical Circuits. (2004) Pearson.
- Eide, A. Engineering Fundamentals & Problem Solving. (2002) McGraw Hill
- MathWorks. MATLAB Reference Material Version R2000a. (2007) MathWorks

I would like to give special thanks to my students and colleagues for their valued contributions in making this material a more effective learning tool.

I invite the reader to forward any corrections, additional topics, examples and problems to me for future updates.

Thanks,

*Izad Khormae*

[www.EngrCS.com](http://www.EngrCS.com)

*Copyright 2010 Izad Khormae, All Rights Reserved.*

## Table of Contents

<b>Chapter 1. Introduction</b>	<b>5</b>
1.1 Key Concepts and Overview	5
1.2. Engineering and Computer Science (ECS)	6
1.3. Education	7
1.4. Careers	9
1.5. Achievements and Benefits	11
1.6. Societies and Certification	13
1.7. Key Success Factors	14
1.8. Patents	15
1.9. Code of Ethics	16
1.10. Additional Resources	21
1.11. Problems	22
<b>Chapter 2. Teamwork and Communication</b>	<b>24</b>
2.1. Key Concepts and Overview	24
2.2. Thinking Modes	25
2.3. Thinking Mode Assessment Based on Herrmann Whole Brain Model	28
2.4. Teamwork	30
2.5. Communication	32
2.6. Additional Resources	36
2.7. Problems	37
<b>Chapter 3. Creative Problem Solving</b>	<b>38</b>
3.1. Key Concepts and Overview	38
3.2. Creative Problem-Solving Process	39
3.3. Step 1 - Customer Issues/Needs Identification	41
3.4. Step2 - Problem Definition	42
3.5 - Step 3 - Idea Generation	44
3.6. Step 4 - Optimal Solution Selection	47
3.7. Step 5 - Solution Implementation	51
3.8. Additional Resources	52
3.9. Problems	53
<b>Chapter 4. Electrical Circuits</b>	<b>54</b>
4.1. Key Concepts and Overview	54
4.2. Charge, Current and Voltage	55
4.3. Circuit Model	57
4.4. Power Sources	59
4.5. Circuit Simplifications	68
4.6. Kirchhoff's Laws and DC Circuit Analysis	71
4.7. Capacitor, C	79
4.8. Inductor, L	81
4.9. Transformers	83
4.10. Additional Resources	84
4.11. Problems	85
<b>Chapter 5. Digital Logic</b>	<b>89</b>
5.1. Key Concepts and Overview	89
5.2. Digital vs. Analog	90
5.3. Digital Design Overview	92

5.4. Binary Number Systems .....	94
5.5. Binary Algebra.....	97
5.6. Standard Logic Gates .....	99
5.7. Simple Input and Output Configurations.....	101
5.8. Logic Design Examples.....	103
5.9. Beyond Standard Logic.....	110
5.10. Additional Resources .....	113
5.11. Problems .....	114
<b>Chapter 6. Computer Architecture and Programming Fundamentals .....</b>	<b>115</b>
6.1. Key Concepts and Overview.....	115
6.2. Computer Architecture .....	116
6.3. Programming Levels .....	119
6.4. Software Development Steps .....	120
6.5. Development Environment Interface and Structure.....	122
6.6. Using the MATLAB Command Window.....	126
6.7. Creating and Editing M-files.....	128
6.8. MATLAB Arithmetic and Logic Operators .....	130
6.9. MATLAB Data Flow Controls .....	133
6.10. Additional Resources .....	142
6.11. Problems .....	143
<b>Chapter 7. Mathematical Concepts.....</b>	<b>145</b>
7.1. Key Concepts and Overview.....	145
7.2. Matrices.....	146
7.3. MATLAB MATRIX OPERATIONS .....	149
7.4. Trigonometry .....	152
7.5. MATLAB Trigonometry Operations.....	158
7.6. Complex Numbers .....	163
7.7. MATLAB Complex Number Operations.....	165
7.8. MATLAB Exploration.....	166
7.9. Additional Resources .....	167
7.10. Problems .....	168
<b>Appendix A. MATLAB and SCILAB Differences .....</b>	<b>170</b>
<b>Appendix B. Additional Resources.....</b>	<b>171</b>

## **Chapter 1. Introduction**

### **1.1 Key Concepts and Overview**

- ❖ Engineering and Computer Science Professions
- ❖ Education
- ❖ Careers
- ❖ Achievements and Benefits
- ❖ Societies and Certifications
- ❖ Key Success Factors
- ❖ Patents
- ❖ The Code of Ethics

## 1.2. Engineering and Computer Science (ECS)

ECS professions are diverse and would require a multi-volume book to fully cover them. The following list of attributes is presented as an overview of ECS:

- ❖ **Definition of Engineering**  
Engineers apply principles of science and mathematics in order to develop innovative solutions to problems.
- ❖ **Electrical Engineering / Computer Science and Engineering**  
The majority of tools and systems in today's world rely on electrical/computer components that require the skills of Electrical Engineers, Computer Scientists and Computer Engineers.

The need for these professionals cuts across all industries without exception. For example, a new airplane has numerous computer and electrical systems that need to be designed and programmed.

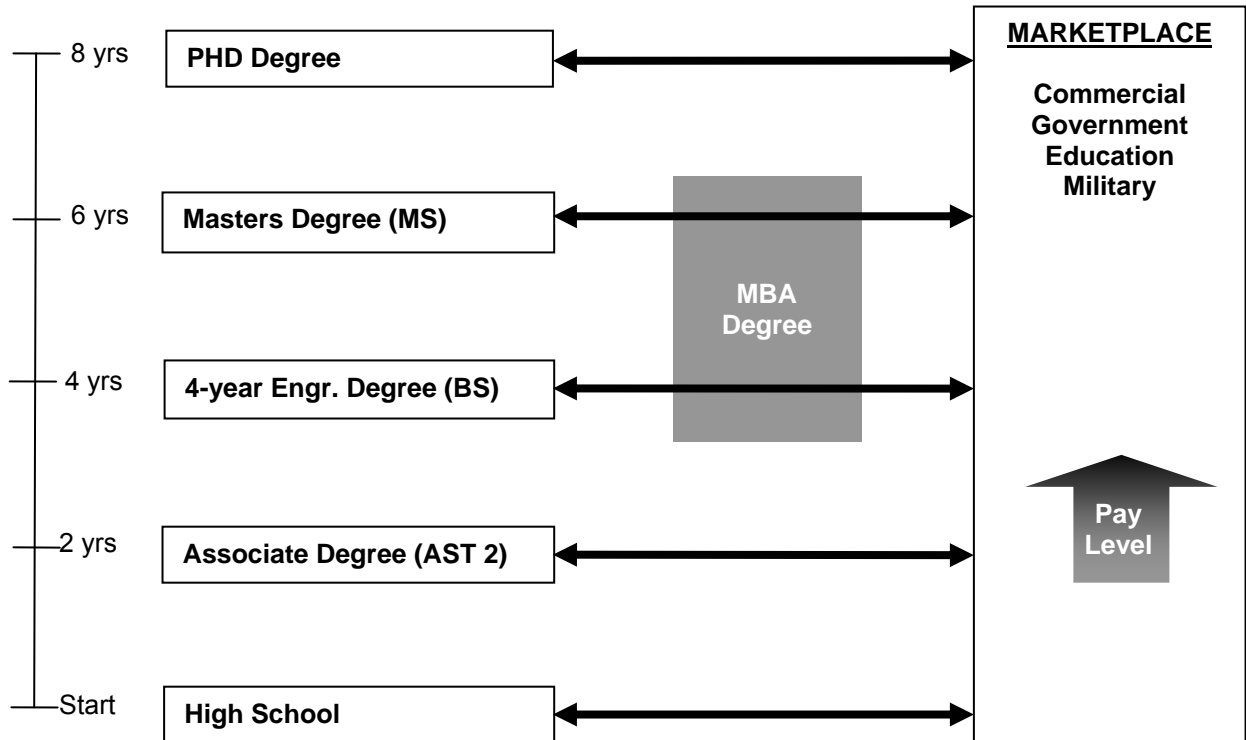
Even household appliances such as a microwaves or refrigerators have computers and electrical systems that need to be designed and programmed.

The challenge to the reader is to think of a self-actuated system that does not require electrical/computer systems.

**Student Exercise** - Can you think of a Company that is not benefiting from Engineering and Computer Science professionals?

### 1.3. Education

The Engineering and Computer Science education can be best described in two-year intervals after high school completion. The following diagram shows the degree attained after approximately each two-year study and entry into the marketplace:



The program at a community college such as Clark covers the first two years of post high school studies resulting in an Associate Degree. At Clark College, the Associate Degree designed for the ECS area is referred to as AST2. Below are the typical course requirements for the first two years of Computer Science, Electrical Engineering and Computer Engineering Degrees:

- Communication, Social Science and Humanities (10-20 Credits)
- Mathematics Courses
  - College Algebra, College Trigonometry
  - Calculus I, II, III, IV
  - Differential Equations and Linear Algebra
  - Discrete Math (Computer Science only)
- Engineering Physics I, II and III
- Introduction to ECSE
- Electrical Circuits and Signal Processing I, II, III –(Electrical and Computer Engineering Only)

- Digital Logic Design
- Microprocessor Design
- Introduction to Programming
- Object Oriented Programming (Computer Science Only)
- Data Structure I, II (Computer Science Only)
- Programming Tools (Computer Science Only)

❖ Benefits of attending Community College

After completing high school, students have the option of attending university or community college. The following list outlines some of the key benefits of community college studies:

- Proven success in the Engineering Transfer Program
- Smaller class size and easy access to professors
- Lower cost – typically half the cost of university programs
- Local and in-person classes - no need to relocate
- Extensive hands-on engineering project opportunities

❖ Common Destination Universities

After completing the first two years of studies in community college, students can transfer as juniors into a four year university to complete a Bachelor's Degree. The top transfer universities are listed in order of the number of student transfers from Clark College:

- Washington State University – Vancouver
- Portland State University
- University of Washington – Seattle
- Washington State University – Pullman
- University of Portland



## 1.4. Careers

Careers in ECS are typically defined based on technology, location in the life cycle, or a mix of the two attributes. The following list outlines some of the technologies, relationships and product development life cycles for these three closely-related fields:

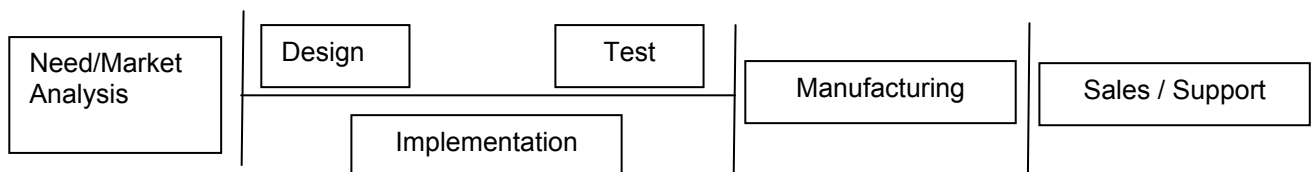
### ❖ Technology Fields

Here are the major technology categories for each of the three fields which are sometimes used to define ECS positions:

- Electrical Engineering
  - Power and Energy Systems
  - Communication Systems
  - Magnetic Fields
  - Semiconductors
  - Electronics
  - Signal Processing
  - Biomedical
  - Control Systems
  - ...
- Computer Engineering
  - Computer Systems
  - Networking
  - Embedded Systems
  - Automation/Robotics
  - Artificial Intelligence
  - Biomedical
  - ...
- Computer Science
  - Application
  - Networking
  - Embedded System
  - Artificial Intelligence
  - Gaming
  - Automation/Robotics
  - ...

### ❖ Life Cycles

Product and Service development life cycle is another way for some organizations to define ECS positions. In other words, positions are defined based on their roles within the product development life cycles. A typical product development life cycle can be outlined as:

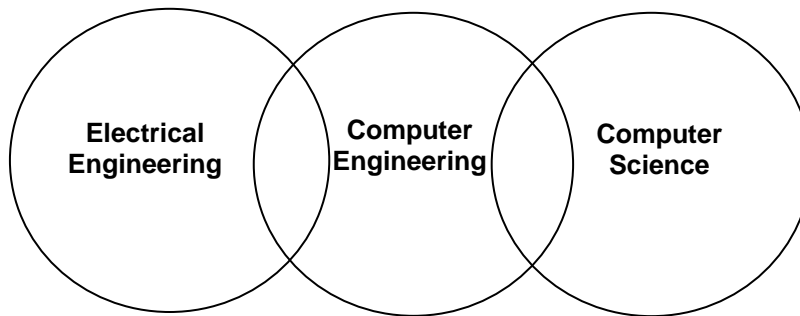


ECS positions, with focus on specific parts of the life cycle, are titled as:

- Design Engineer
- Development Engineer
- Test Engineer
- Process Engineer
- Application Engineer
- Sales Engineer
- Support Engineer
- Field Engineer

❖ Relationships Among the ECS Fields

There are overlaps among Electrical Engineering, Computer Engineering and Computer Science. It is common that a professional in one field engages in cross discipline projects. For example if the robotic project is being worked on then all three disciplines will be involved. The following Venn Diagram depicts the overlap and relationships among the three fields:



## 1.5. Achievements and Benefits

In this Section, an attempt is made to reflect on past achievements and allow for students to imagine the possibilities of the next 10 years. Finally, careers are selected based on benefits of the career and one's interest which are discussed at the end of this Section.

- ❖ Greatest Engineering and Computer Science (ECS) Achievements in the Past 100 Years
  - #10 - Space Explorations
  - # 9 - High Performance Material
  - # 8 - Genetic and Bio Engineering
  - # 7 - Health and Medical Technology
  - # 6 - Transportation (Automobile, Airplane)
  - # 5 - Communication Technology (Phone, Radio/Television, Internet)
  - # 4 - Electronics and Computers
  - # 3 - Air Conditioning and Refrigeration
  - # 2 - Agricultural Mechanization
  - # 1 - Easy Access to Safe and Drinkable Water
  
- ❖ Expected top ten Engineering and Computer Science (ECS) Achievements in the Next Decade

**Student Exercise** – List the top ten ECS Achievements in the Next Decade: :

- #10 -
- # 9 -
- # 8 -
- # 7 -
- # 6 -
- # 5 -
- # 4 -
- # 3 -
- # 2 -
- # 1 -

- ❖ Benefits of an Engineering Career:
  - Broad Range of Opportunities and Growth Potential – ECS education is a great foundation for a broad range of technical and management positions. ECS professionals are problem-solvers and although most start solving technical problems, over time, their careers may take many directions. ECS educated professionals are found working as Designers, Developers, Managers and even Chief Executive Officers (CEO).
  - Intellectually Challenging Work – ECS professionals are employed to solve problems that have not been previously solved. This means the projects are new or a variation of a past problem but always interesting. Additionally, opportunities are limitless and only gated by the willingness and ability of the professional.
  - Adding Value to Society and Being Financially Rewarded - Engineers comprise about 1% of population but directly affect over 90% of society's production capacity. It is impossible to think of a product or service that has not benefited from the contributions of Engineers and Computer Scientists.

An ECS career is financially rewarding, with a starting salary typically ranging from \$40,000 to \$100,000 per year. Additionally, the work environment is safe, comfortable and staff supported.

**Student Exercise** – What are the top three benefits that you desire from your career? Does an ECS career provide you with these desired benefits?

## 1.6. Societies and Certification

This Section covers the need to get involved in Professional Societies/Clubs, and introduces the Professional Engineer (PE) certification.

### ❖ Professional Societies/Clubs

Societies and Clubs help ECS students share ideas, get hands-on experience and meet professionals. Additionally, the experience gained in leadership and technical projects plays an important role in finding and getting your first internship or job.

Students are encouraged to consider joining and becoming active in one or more of the following groups:

- Clark College Computing and Electrical (CE) Club
- Institute of Electrical and Electronics Engineering (IEEE)
- Association for Computing Machinery (ACM)

### ❖ Professional Engineer (PE) Certification

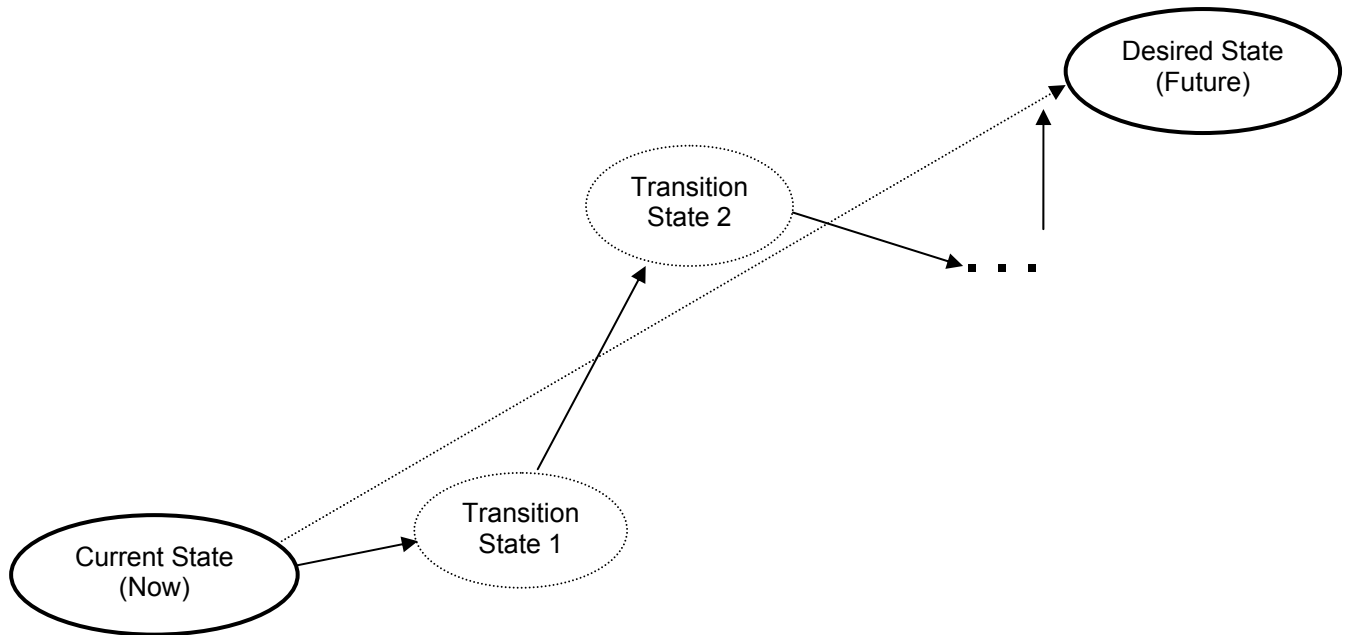
Engineering may obtain a Professional Engineer (PE) Certification by completing the following requirements:

- Completion of an engineering degree from an ABET-credited College
- Passing the “Engineer In Training” (EIT) Exam
- Completing four years of practice under the supervision of a professional engineer
- Passing the “Principles and Practice of Engineering” PE examination

PE Certification is important for Engineers who will be working in areas directly effecting public safety such as Civil and Structural Engineers who are in private practice.

## 1.7. Key Success Factors

Success is not an accident; rather it is the result of careful planning and hard work. Successful people know that success is driven by planning, persistence and focus on goals (Desired State). Below is a formalized model for setting up long-term goals and the intermediate steps needed in order to achieve goals:



Typically, the above model is referred to as the Current-Desired State Model or Diagram. This tool helps focus the person or the organization on long-term goals and ensures that all activities are evaluated on immediate values and contributions to the long-term goal (Desired State).

Once the long-term plan is developed, the next step is to consider the factors for successful execution. The most important factor is a positive attitude, which simply means seeing success as reaffirmation of the direction, and failure as the opportunity to learn, adjust plan and try again.

The next item to consider is “working smart”, which means when investing resources on a task, consider the impact on your success. If the task is not producing the desired result, then adjust approach to improve effectiveness. We all have limited time and resources, so it is important to maximize their positive impacts.

## 1.8. Patents

Typically, engineers are either given or find problems that, if solved, add value to society. The Patent Office was created to protect the right of inventors to exclusively benefit from their inventions.

Any idea, process, procedure, system or design may be patented as long as it passes the following two tests:

- 1) Novel  
The “novel” test is passed if the topic being patented is not described in an existing document or patent.
- 2) Non-Obvious  
The “non-obvious” test is passed if the topic being patented cannot be described by multiple existing documents or patents.

The Patent Office is given the responsibility to evaluate and approve patents. When applying for a patent, it is important to answer the following questions:

- What is the problem and who will benefit from the proposed solution?
- What are the new inventions being claimed by this patent? (Only new ideas in the solutions are being claimed are referred to as the claims)
- What are the existing patents that relate to the claim(s) being made?

The law allows for protection other types of intellectual property such as:

- Copyright ©  
Any written material can be protected through Copyright process.
- Trademark  
Images associated with a product can be protected through Trademark process.
- Servicemark  
Images associated with a service can be protected through Servicemark process.

More information on the patent application is available at the Patent Office online website at [www.uspto.gov](http://www.uspto.gov).

## 1.9. Code of Ethics

Each profession is guided by a code of ethics which is intended to help members conduct their professional duties in an ethical way. Ethical performance benefits the profession and its members in the long-run.

- ❖ Definition of Ethics according to dictionary.com  
**ethics** (*used with a sing. or pl. verb*) The rules or standards governing the conduct of a person or the members of a profession:
  - A set of principles of right conduct.
  - A theory or a system of moral values: “An ethic of service is at war with a craving for gain” (Gregg Easterbrook).
  
- ❖ Stakeholders  
Stakeholders are individuals and groups who are in the best position to impact the success of program and also they are the ones who the program most directly impacts. In a typical program or project, the following groups are typically included as stakeholders with respect to ethics:
  - Self
  - Employer
  - Public
  - Client
  - Users

Here, the National Society of Professional Engineers (NSPE) Code of Ethics for Engineers (Revised January 2003/ Reprint permission granted on 1/2004) is used as an example of an engineering code of ethics.

- ❖ Preamble  
Engineering is an important and learned profession. As members of this profession, engineers are expected to exhibit the highest standards of honesty and integrity. Engineering has a direct and vital impact on the quality of life for all people. Accordingly, the services provided by engineers require honesty, impartiality, fairness, and equity, and must be dedicated to the protection of the public health, safety, and welfare. Engineers must perform under a standard of professional behavior that requires adherence to the highest principles of ethical conduct.
  
- ❖ Fundamental Canons  
Engineers in the fulfillment of their professional duties shall:
  - Hold paramount the safety, health and welfare of the public.
  - Perform services only in areas of their competence.
  - Issue public statements only in an objective and truthful manner.
  - Act for each employer or client as faithful agents or trustees.
  - Avoid deceptive acts.
  - Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession.



❖ Rules of Practice

- 1) Engineers shall hold paramount the safety, health, and welfare of the public.
  - a) If engineers' judgment is overruled under circumstances that endanger life or property, they shall notify their employer or client and such other authority as may be appropriate.
  - b) Engineers shall approve only those engineering documents that are in conformity with applicable standards.
  - c) Engineers shall not reveal facts, data, or information without the prior consent of the client or employer except as authorized or required by law or this Code.
  - d) Engineers shall not permit the use of their name or associate in business ventures with any person or firm that they believe are engaged in fraudulent or dishonest enterprise.
  - e) Engineers shall not aid or abet the unlawful practice of engineering by a person or firm.
  - f) Engineers having knowledge of any alleged violation of this Code shall report thereon to appropriate professional bodies and, when relevant, also to public authorities, and cooperate with the proper authorities in furnishing such information or assistance as may be required.
- 2) Engineers shall perform services only in the areas of their competence.
  - a) Engineers shall undertake assignments only when qualified by education or experience in the specific technical fields involved.
  - b) Engineers shall not affix their signatures to any plans or documents dealing with subject matter in which they lack competence, nor to any plan or document not prepared under their direction and control.
  - c) Engineers may accept assignments and assume responsibility for coordination of an entire project and sign and seal the engineering documents for the entire project, provided that each technical segment is signed and sealed only by the qualified engineers who prepared the segment.
- 3) Engineers shall issue public statements only in an objective and truthful manner.
  - a) Engineers shall be objective and truthful in professional reports, statements, or testimony. They shall include all relevant and pertinent information in such reports, statements, or testimony, which should bear the date indicating when it was current.
  - b) Engineers may express publicly technical opinions that are founded upon knowledge of the facts and competence in the subject matter.
  - c) Engineers shall issue no statements, criticisms, or arguments on technical matters that are inspired or paid for by interested parties, unless they have prefaced their comments by explicitly identifying the interested parties on whose behalf they are speaking and by revealing the existence of any interest the engineers may have in the matters.
- 4) Engineers shall act for each employer or client as faithful agents or trustees.
  - a) Engineers shall disclose all known or potential conflicts of interest that could influence or appear to influence their judgment or the quality of their services.
  - b) Engineers shall not accept compensation, financial or otherwise, from more than one party for services on the same project, or for services pertaining to the same project, unless the circumstances are fully disclosed and agreed to by all interested parties.
  - c) Engineers shall not solicit or accept financial or other valuable consideration, directly or indirectly, from outside agents in connection with the work for which they are responsible.
  - d) Engineers in public service as members, advisors, or employees of a governmental or quasigovernmental body or department shall not participate in decisions with respect to services solicited or provided by them or their organizations in private or public engineering practice.

- e) Engineers shall not solicit or accept a contract from a governmental body on which a principal or officer of their organization serves as a member.
- 5) Engineers shall avoid deceptive acts.
- a) Engineers shall not falsify their qualifications or permit misrepresentation of their or their associates' qualifications. They shall not misrepresent or exaggerate their responsibility in or for the subject matter of prior assignments. Brochures or other presentations incident to the solicitation of employment shall not misrepresent pertinent facts concerning employers, employees, associates, joint ventures, or past accomplishments.
  - b) Engineers shall not offer, give, solicit or receive, either directly or indirectly, any contribution to influence the award of a contract by public authority, or which may be reasonably construed by the public as having the effect of intent to influencing the awarding of a contract. They shall not offer any gift or other valuable consideration in order to secure work. They shall not pay a commission, percentage, or brokerage fee in order to secure work, except to a bona fide employee or bona fide established commercial or marketing agencies retained by them.

❖ Professional Obligations

- 1) Engineers shall be guided in all their relations by the highest standards of honesty and integrity.
- a) Engineers shall acknowledge their errors and shall not distort or alter the facts.
  - b) Engineers shall advise their clients or employers when they believe a project will not be successful.
  - c) Engineers shall not accept outside employment to the detriment of their regular work or interest. Before accepting any outside engineering employment they will notify their employers.
  - d) Engineers shall not attempt to attract an engineer from another employer by false or misleading pretenses.
  - e) Engineers shall not promote their own interest at the expense of the dignity and integrity of the profession.
- 2) Engineers shall at all times strive to serve the public interest.
- a) Engineers shall seek opportunities to participate in civic affairs; career guidance for youths; and work for the advancement of the safety, health, and well-being of their community.
  - b) Engineers shall not complete, sign, or seal plans and/or specifications that are not in conformity with applicable engineering standards. If the client or employer insists on such unprofessional conduct, they shall notify the proper authorities and withdraw from further service on the project.
  - c) Engineers shall endeavor to extend public knowledge and appreciation of engineering and its achievements.
- 3) Engineers shall avoid all conduct or practice that deceives the public.
- a) Engineers shall avoid the use of statements containing a material misrepresentation of fact or omitting a material fact.
  - b) Consistent with the foregoing, engineers may advertise for recruitment of personnel.
  - c) Consistent with the foregoing, engineers may prepare articles for the lay or technical press, but such articles shall not imply credit to the author for work performed by others.
- 4) Engineers shall not disclose, without consent, confidential information concerning the business affairs or technical processes of any present or former client or employer, or public body on which they serve.

- a) Engineers shall not, without the consent of all interested parties, promote or arrange for new employment or practice in connection with a specific project for which the engineer has gained particular and specialized knowledge.
  - b) Engineers shall not, without the consent of all interested parties, participate in or represent an adversary interest in connection with a specific project or proceeding in which the engineer has gained particular specialized knowledge on behalf of a former client or employer.
- 5) Engineers shall not be influenced in their professional duties by conflicting interests.
- a) Engineers shall not accept financial or other considerations, including free engineering designs, from material or equipment suppliers for specifying their product.
  - b) Engineers shall not accept commissions or allowances, directly or indirectly, from contractors or other parties dealing with clients or employers of the engineer in connection with work for which the engineer is responsible.
- 6) Engineers shall not attempt to obtain employment or advancement or professional engagements by untruthfully criticizing other engineers, or by other improper or questionable methods.
- a) Engineers shall not request, propose, or accept a commission on a contingent basis under circumstances in which their judgment may be compromised.
  - b) Engineers in salaried positions shall accept part-time engineering work only to the extent consistent with policies of the employer and in accordance with ethical considerations.
  - c) Engineers shall not, without consent, use equipment, supplies, laboratory, or office facilities of an employer to carry on outside private practice.
- 7) Engineers shall not attempt to injure, maliciously or falsely, directly or indirectly, the professional reputation, prospects, practice, or employment of other engineers. Engineers who believe others are guilty of unethical or illegal practice shall present such information to the proper authority for action.
- a) Engineers in private practice shall not review the work of another engineer for the same client, except with the knowledge of such engineer, or unless the connection of such engineer with the work has been terminated.
  - b) Engineers in governmental, industrial, or educational employ are entitled to review and evaluate the work of other engineers when so required by their employment duties.
  - c) Engineers in sales or industrial employ are entitled to make engineering comparisons of represented products with products of other suppliers.
- 8) Engineers shall accept personal responsibility for their professional activities, provided, however, those engineers may seek indemnification for services arising out of their practice for other than gross negligence, where the engineer's interests cannot otherwise be protected.
- a) Engineers shall conform to state registration laws in the practice of engineering.
  - b) Engineers shall not use association with a non-engineer, a corporation, or partnership as a "cloak" for unethical acts.
- 9) Engineers shall give credit for engineering work to those to whom credit is due, and will recognize the proprietary interests of others.
- a) Engineers shall, whenever possible, name the person or persons who may be individually responsible for designs, inventions, writings, or other accomplishments.
  - b) Engineers using designs supplied by a client recognize that the designs remain the property of the client and may not be duplicated by the engineer for others without express permission.
  - c) Engineers, before undertaking work for others in connection with which the engineer may make improvements, plans, designs, inventions, or other records that may justify copyrights or patents, should enter into a positive agreement regarding ownership.

- d) Engineers' designs, data, records, and notes referring exclusively to an employer's work are the employer's property. The employer should indemnify the engineer for use of the information for any purpose other than the original purpose.
- e) Engineers shall continue their professional development throughout their careers and should keep current in their specialty fields by engaging in professional practice, participating in continuing education courses, reading in the technical literature, and attending professional meetings and seminars.

### 1.10. Additional Resources

- ❖ Wakerly, I. Digital Design. (2006) Prentice Hall
- ❖ Katz, R. Contemporary Logic Design. (2005) Pearson.
- ❖ Lumsdaine, E. Creative Problem Solving and Engineering. (1999) Prentice Hall.
- ❖ Sandige, R. Digital Design Essentials. (2002) Prentice Hall.
- ❖ Nilsson, J. Electrical Circuits. (2004) Pearson.
- ❖ Eide, A. Engineering Fundamentals & Problem Solving. (2002) McGraw Hill
- ❖ MathWorks. MATLAB Reference Material Version R2000a. (2007) MathWorks

## 1.11. Problems

---

1. Use the job listing services online to find five open positions that would be of interest to you and compatible with the degree you are planning to pursue. Document the title, hiring company, position description and candidate qualifications.

*Note: Be prepared to discuss and present in-class.*

---

2. Identify the top three engineering problems that you expect will be solved in the next ten years. Explain your reasons for the selections.

*Note: Be prepared to discuss and present in-class.*

---

3. Develop a patent application for one of your new ideas. A new idea is defined as the idea for a product, service or improvement that is currently not available on the market. Your report is expected to include the following:

- 1) Problem definition
  - 2) Description of similar products on the market that may solve the problem, and comparative advantages/disadvantages of your product idea.
  - 3) Include the following information:
    - a) Claims (specific inventions)
    - b) Research to support your invention
    - c) An instantiation of the Claims (product utilizing the invention)
    - d) List of patent numbers and titles for the top three relevant patents
- 

4. Your company bids on a government job and gets it. You are an engineer at this company. Further, you know your company is making above market profit on the government contract. What is your ethical responsibility and why? Base your answer on the NSPE Code of Ethics.

---

5. You are a project engineer for a heart monitor development. You have been reviewing the test data that shows 2 out of 100,000 tests failed. Your manager thinks .002% is a great test result and the product should be released to production. What is your ethical responsibility and why? Base your answer on the NSPE Code of Ethics.

---

6. Your company has a contract to build a bridge. Material specified in the contract is no longer readily available so your company substitutes another type of material. What are your disclosure responsibilities at this point and to whom? Base your answer on the NSPE Code of Ethics.

---

7. Develop a five-year-plan for yourself and document the plan using the Current-Desired diagram. For this exercise, show one transition state for each year of the plan.

---

8. Complete your study plan for the next two years and review it with your department academic advisor.

---

9. Associations and clubs are a critical success factors in your education and/or career. Identify five clubs or associations that you feel would best support your education and/or career. Next, select one as the most important one and explain your reason for the selection.

---

10. Name 5 specializations within Electrical and Computer fields.

---

11. What are patent claims? Explain their significance and provide a claim example?

---

---

12. What are the five fundamental canons of NSPE code of Ethics?

---

13. Name the top three areas of research/trends affecting the future opportunities in Electrical and Computing fields. Explain the reason for your selection.

---

14. In regard to Code of Ethics, what is definition of a stakeholder and which groups can be considered stakeholders?

## **Chapter 2. Teamwork and Communication**

### **2.1. Key Concepts and Overview**

- ❖ Thinking Modes or Styles
- ❖ Teamwork
- ❖ Communication



## 2.2. Thinking Modes

Thinking modes also referred to as the mental model, thinking preferences or thinking styles is the foundation by which we make decisions, plan, communicate and execute tasks. Organizations with employees who have shared thinking styles are the most effective organizations, since they have improved communication and teamwork. Their effectiveness results from the fact that a shared model is the process we use to understand our environment and decision making-processes.

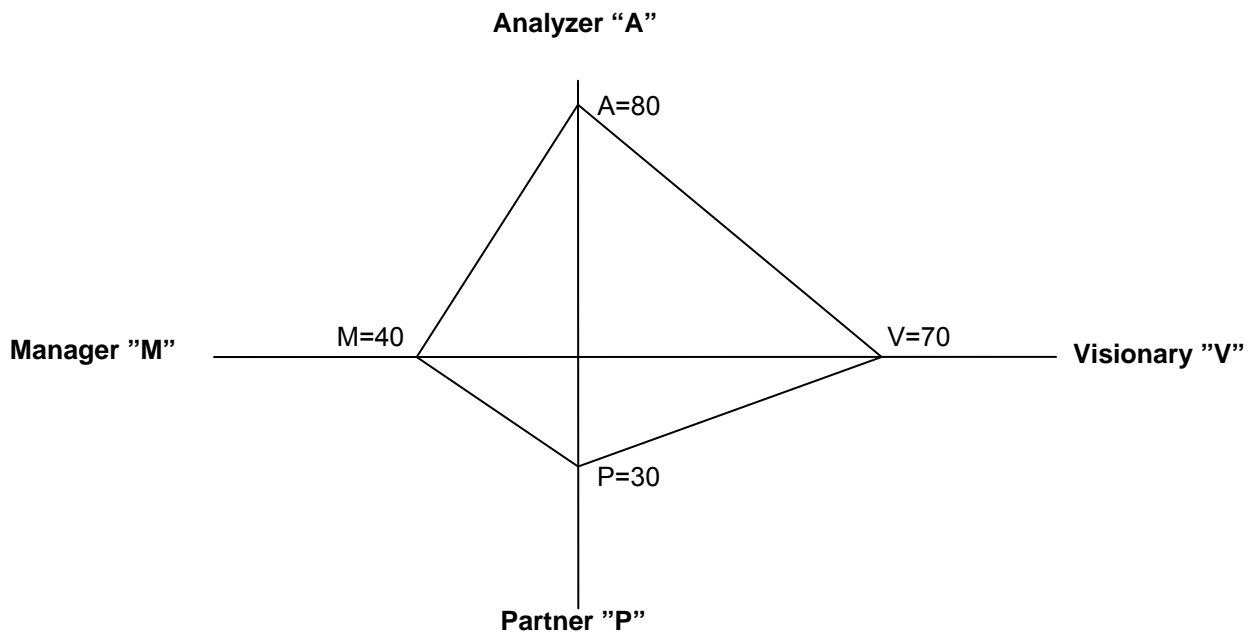
Although there are a large variety of evaluation tools in the market for thinking modes evaluation, this text introduces a variation of the Herrmann Brain Domain Model (HBDM). These types of evaluation are most commonly referred to as “personality tests” and are commonly used by employers to assess teamwork, job performance and communication preference of their employees. One of the most common personality tests used in business is call Myers-Briggs personality test.

The reason for the presentation of the Herrmann Model here is to provide a tool for evaluating self and teammate preference, to improve communication, and by extension, teamwork and performance. Another benefit is to have sufficient information to create a team with diverse ways of thinking, since it has been shown that these teams outperform less diverse-thinking teams.

The Herrmann Brian Model (HBM) attempts to identify an individual’s preferred thinking mode in accordance with four thinking styles (also called thinking modes preferences). The four thinking modes are:

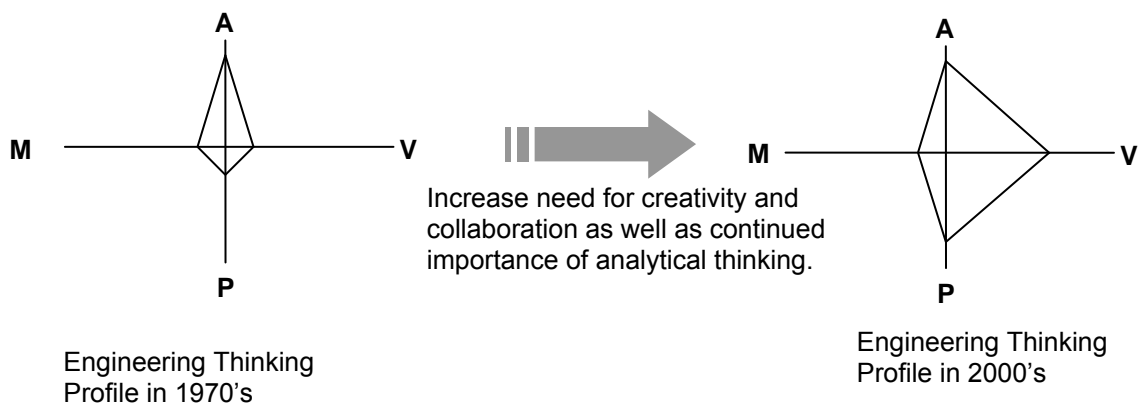
- Analyzer “A”
  - Requires facts and data that fits together logically
  - Likes data analysis
  - Learns and makes decision based on information from experts and proven facts
- Manager “M”
  - Requires a proven and detailed execution plan that minimizes risks
  - Likes low-risk activities with clear step-by-step plan from beginning to end
  - Learns and makes decisions based on testing results and proven past experience
- Partner “P”
  - Requires human interaction and knowledge that everyone is taken care
  - Likes to work on relationship development and help with environment/community/politics
  - Learns and makes decision based on feeling, personal experiences and in-person feedback from others
- Visionary “V”
  - Requires change and innovation and loses interest once the project is understood
  - Likes to work based on inspirations and insights
  - Learns by conceptualizing possibilities and exploring alternative to known solutions and processes.

An Individual may be stronger in one or more of the quadrants, but most humans move among the different thinking modes. Typically, the assessment is displayed as a radar chart. For example if a person completes the evaluation tool that will be presented later with A=80, M=40, P=30 and V=70, then the evaluation may be plotted as the following Radar Chart (Radar Chart is a chart with multiple axes):

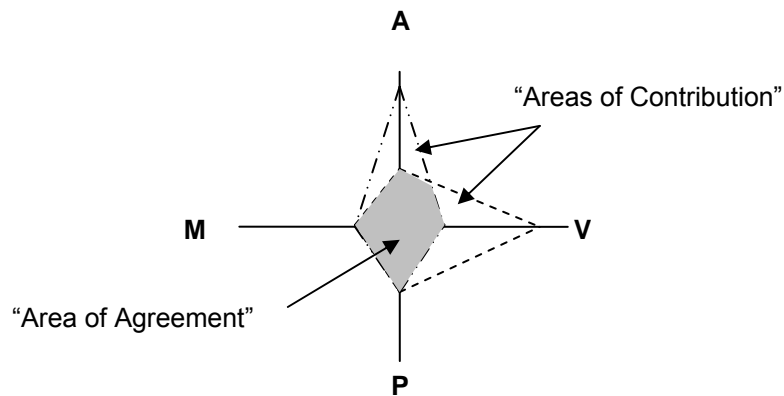


Based on the above Chart, we can state that this individual is highly creative (V=70) and analytical (A=80), but not as interested in managing a project plan or working with others. So it is important that other teammates have stronger Partner and Manager attributes for optimal performance.

Although such an individual would have had great success as an engineer in the past, in today's collaborative environment, this individual needs to work on his collaboration skills. Over a 30-year period the thinking attributes of a successful engineer has changed as shown below:



Thinking model is also an important consideration when selecting team members, since it allows for understanding of thinking diversity. The following diagram shows overlap of two persons' thinking profile:



The area of overlap is called “Areas of Agreement”. As the name implies, when both are thinking in the “Area of Agreement”, they will reach the same conclusion given the same facts. Although some overlap is required to encourage team work, it does not represent any contribution in creating new ideas.

The areas of non-overlap are called “Areas of Contribution” since each person will have a different perspective and is likely to have a different solution given the same facts. Therefore, teams operating in this region have a higher probability of generating creative solutions with broader effectiveness.

When selecting team members, it is important to balance the size of areas of agreement and the size of areas of contribution. One allows for improved teamwork while the other allows for increased creativity and diversity in thinking.

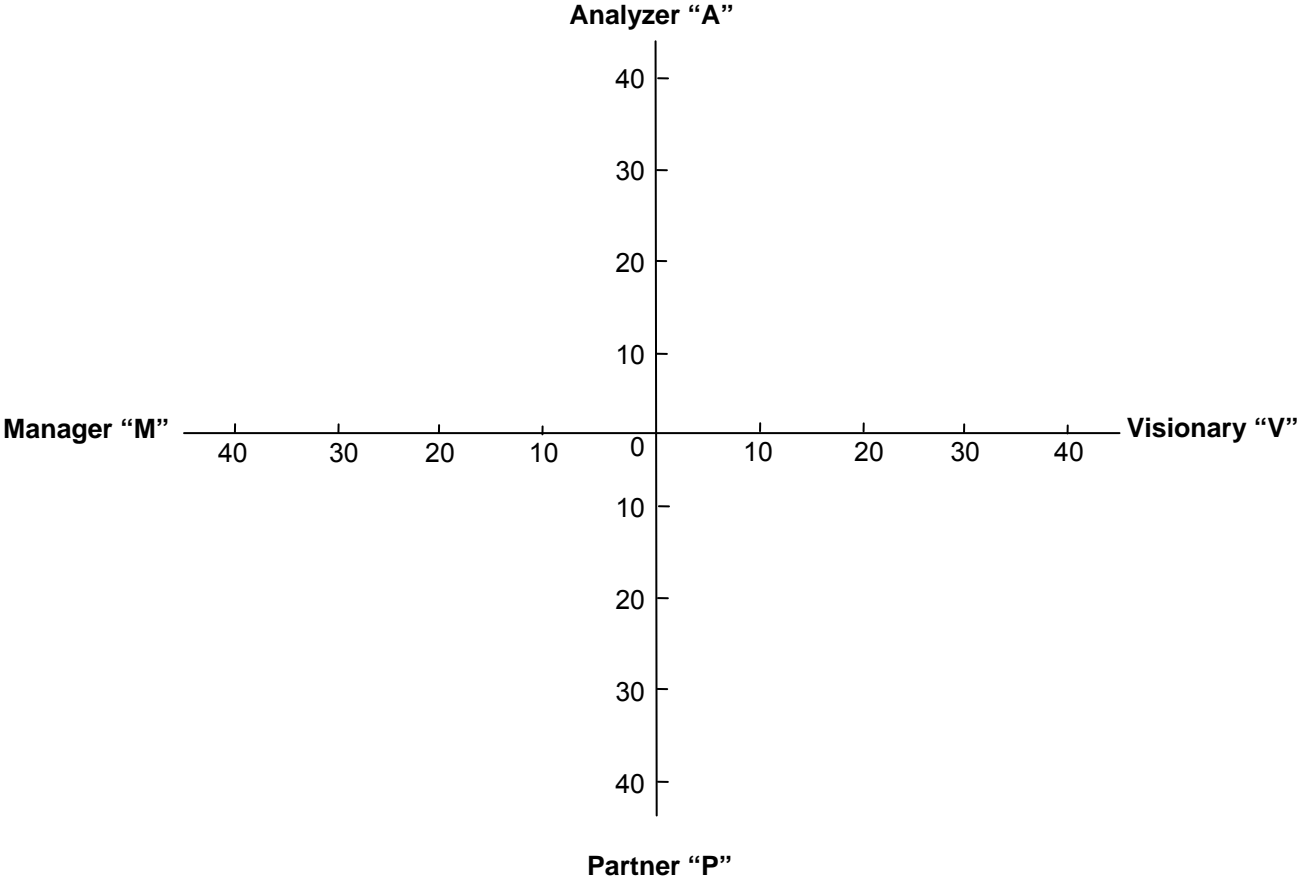
### 2.3. Thinking Mode Assessment Based on Herrmann Whole Brain Model

The following Assessment tool is designed to assess the thinking preference of the user by completing the tool and drawing the resulting radar chart that may be analyzed. This tool is adapted from pre-existing tool (unknown original source).

**STEP 1. For each statement, rank responses from 4 (most likely) to 1 (least likely).**

Statements	A Rank	M Rank	P Rank	V Rank
1. In a project, you prefer to: A. Work with data and facts; M. Develop a plan and schedule; P. Meet and discuss ideas with others; V. Look for the broader view and next new idea.				
2. You best learn new ideas by: A. Applying them to actual situations; M. Conducting careful analysis; P. Discussing with others; V. Thinking of new ways to use the idea.				
3. When working on a problem, you: A. Organize the material logically but not too much detail; M. Organize material neatly and pay attention to detail. P. Look for impacts on others and society; V. Explore new ways to solve the problem.				
4. In studying new material, you prefer to: A. Read the text book and listen to fact-based lecture. M. Test processes and procedures to find the problems; P. Read the introduction and overview to understand the purpose; V. Explore possibilities by asking 'what-if' questions.				
5. After solving a problem, you: A. Think through ideas rationally; M. Find practical uses of the knowledge learned; P. share with others; V. Appreciate the elegance of the solution and explore other solutions.				
6. The most important item in a new article for you is: A. Its basis in fact; M. Validity of the idea based on past data; P. Practicality of the idea; V. The relevance of the idea to your opinion.				
7. In project execution, you prefer to: A. Make up a theory and then test it; M. Develop a plan and execute the project according to plan; P. Do experiments and observe the results; V. Experiment and explore ideas and possibilities.				
8. You are most influences by: A. Ideas that are based on facts and logic; M. The conviction and force used to express the idea; P. closeness of idea to your opinion; V. Ideas that are broad in nature and forward looking.				
9. In your decision making process, you rely most on: A. Reality and facts; M. Proven track record, detailed studies; P. Discussion with others and their opinions; V. Your own feeling and views.				
<b>Total:</b>				

**STEP 2. Plot your scores on the following Radar Chart:**



**STEP 3. Describe the Chart in Terms of Thinking Preferences**

## 2.4. Teamwork

In the past, it was possible for engineers to work in isolation with little or no interaction with peers, teams, departments and organizations. Increasingly, engineers work as a team in order to accomplish their tasks. In a today's typical project team, members include all business functions (marketing, design, manufacturing, customer support, finance, risk management, regulatory), customers and suppliers.

Success as an engineer relies heavily on the engineer's ability to work effectively in a diverse team. In this section, we will discuss team processes and tools that improve team effectiveness.

❖ Steps in organizing an effective team:

- Select members who complement each other's Thinking Model (Herrmann Brain Model) and project expertise.
- Develop a shared vision as a team. Team members should be visualizing project success similarly.
- Agree on a shared objective, including measures of success and a delivery timeline.
- Generate a shared execution plan that is consistent with the project's vision and objectives. The plan must be agreed upon by all members, and care should be taken to ensure optimum leverage.
  - Roles and responsibility
  - Structure
  - Deliverable timeline
- Evaluate the team performance based on the results. Modify vision, objectives and execution plan to improve performance.

**Student Exercise** - Identify 3-4 other students to form a team by working with your instructor, and use the above steps to organize an effective team.

❖ Team members of a typical engineering project include:

- Customers
- Sales and Marketing
- Design, Development (all disciplines)
- Testing
- Implementation and Manufacturing
- Quality Control
- Regulatory Agencies
- Risk Management
- Investors

❖ Team Development Overview

Team development has been the subject of numerous studies and go back to early days of human development. In recent times, it is understood that teams develop and mature through a well defined development process in four stages.

Although change in team members, environment or project definition may result in the team moving

from one stage to the next or back. The Four Stages of Team Development are Forming → Storming → Norming → Performing. This 4-stage development process was first proposed by Bruce Tuckman in 1965. He believed that all of the four stages are necessary and inevitable in team development and its ability to work on projects as a team.

It is important to know that not all teams reach the performing stage; most teams move up and down the development stages depending on the project challenges and current environment. Below is a more detailed description of the Four Team Development Stages:

- **Forming Stage**  
This is the first stage of team development where team members are attempting to learn about each other. Members work on getting to know some basic information about each other such as contact information, experience, skill and interests. Teams in this stage also attempt to understand or establish acceptable group behavior, purpose and goals.
- **Storming Stage**  
Teams move to Storming from Forming Stage. This transition occurs after team members have learned about each other and now are attempting to test the limit of each other's ability and limitation. This testing process typically leads to more conflicts. For a successful team, this stage will allow the members to learn to deal with conflicts.
- **Norming Stage**  
Upon completion of Storming Stage, team typically moves to the Norming Stage. In this stage, team members already know each other, have established limits and are able to resolve conflicts. For many teams, Norming is the highest level the team will attain. Teams in this stage are developing and adopting team processes that are mutually acceptable and are able to effectively deliver results.
- **Performing Stage**  
Although teams in Norming Stage are delivering results and are effective, it is the rare team that reaches the Performing Stage, and is able to deliver results at levels that are significantly higher than the Norming Stage team. Members of a team at the Performing Stage have accepted each other's strengths and weaknesses. Further, they are able to effortlessly build on each other's strengths and supplement each other's weaknesses.

A team in Performing Stage delivers spectacular results with seeming little effort - much like a perfectly tuned machine.

**Student Exercise** - Work with your Team to identify its current stage of development.

#### ❖ Tools for Managing Teams

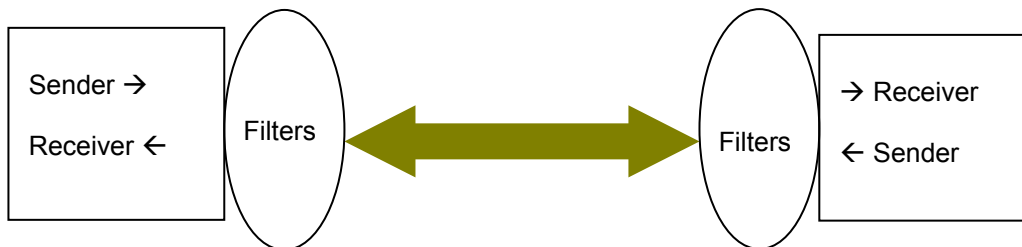
The following tools have proven to be useful in improving the effectiveness of teams. It is recommended that early in your team development, review the list and identify the tools your team will use in the project.

- Team's mission statement and objectives.
- Project plan and timeline.
- Team member role definitions.
- Team ground rules.
- Meeting agendas.
- Task and issues list.
- Team member evaluations (Peers, supervisor, subordinates).

## 2.5. Communication

In all communications, there is a sender and a receiver of information. In successful communications, the impact of information on the receiver is the same as is expected by sender of the information.

In human communications, understanding of information is colored by the biases of the receiver and sender, shown as filters in the following diagram. Additionally the information being transmitted includes the formal part (verbal or written) and informal part (body language and other informal indicators conveying feelings, values, hopes and dreams). Therefore, communication needs to be focused and adapted to the audience for success.



The single biggest challenge in communication is the assumption that communication has taken place. All effective communicators pay special attention to clues of validity of this assumption.

As mentioned earlier, knowing the audience is the key to effective communication. Fortunately, the Herrmann Brain Model is an excellent tool for knowing audience preferences. The following table provides a guide of best approach for the various thinking preferences:

<p><b>Analyzer “A”</b>  <b>”Needs Facts and Logic”</b></p> <ul style="list-style-type: none"> <li>• Facts and Data are supported and documented</li> <li>• Communication is supported by logical use of facts and data</li> </ul>
<p><b>Manager “M”</b>  <b>”Needs Predictability and Low Risk”</b></p> <ul style="list-style-type: none"> <li>• Plan is in detail and has been tested and proven by past experience</li> <li>• Communication is well-organized, detailed and meets the requirements</li> </ul>
<p><b>Visionary “V”</b>  <b>”Needs new challenges and change”</b></p> <ul style="list-style-type: none"> <li>• Plan is new, exciting, outcome is unknown and has not been tried before</li> <li>• Communication must be imaginative, futuristic and emphasize the change/newness</li> </ul>
<p><b>Partner “P”</b>  <b>”Needs relationship and emotion appeal”</b></p> <ul style="list-style-type: none"> <li>• Plan has to be people centric, improve relationships and society</li> <li>• Communication must be playing to emotions and people experiences</li> </ul>



**Student Exercise** - Meet with your team and discuss the team members:

- Thinking preferences
- Technical skills in Electrical, Computing and Mechanical Assembly
- Communication, Teamwork and other relevant skills

At the end of the first meeting, each member is expected to answer the following questions:

- All members' first and last names
- All members' strongest skills
- Gaps in the team
- Team name

❖ There are a variety of communication situations and each requires its own approach to communication. Here are the most common communication categories:

- Negotiation
- Group Presentation
- Written Reports
- General Social or Business Conversation
- Remote (Phone, Online, ...)
- Student Exercise - Identify other communication categories?

In the remainder of this section, we will focus on the Negotiation, Group Presentation and Written Reports.

❖ Negotiation

Negotiation is a form of communication that everyone engages in during the course of life and work activities. In general, there are three types of negotiation based upon the power relationships between parties.

➤ Soft Negotiation

Power Position: One side is giving in; usually due to real or perceived lack of power.

Results: Commonly leads to resentment in the long-term and poisons future relationships.

➤ Hard Negotiation

Power Position: Both parties either have or are perceived to have equal power and typically are focused on the short-term where neither party gives in.

Results: Hard negotiation leads to conflict and counter-productive behaviors but typically exposes areas of conflict.

➤ Principled, interest-based or Win-Win Negotiation

Power Positions: Both parties approach the negotiation as equals with interest in results that maximize total benefits.

Results: These types of negotiations require a high level of trust and time to complete. All issues are decided on merit and to maximize benefits to both parties. This fact leads to both parties working toward a superior outcome that will benefit everyone concerned. Principled Negotiation benefits both parties at a higher level in the long-term as compared to other types of negotiations.

❖ Group Presentation

Group Presentation is one area of communication that most professionals do not engage in as often as other forms of communication. But it is still important for a successful education and career. Here

are some key considerations in group presentations:

- Know your audience's interests and issues. Create your presentation with the audience in mind.
- Command attention and get the audience on your side.
  - Introduce yourself
  - State the purpose of the presentation in terms that capture the interest of audience
  - Acknowledge the needs and expectations of the audience
- Audience will be looking for non-verbal cues from the moment you start.
  - Stand erect
  - Seek eye contact
  - Speak clearly with calm authority
  - Project energy, enthusiasm and competence
  - Use visual aids only to emphasize key points
- Audience can remember 3 to 5 main points only if they are repeated and reinforced
  - Focus on 3-5 main points
  - Keep reinforcing key points in your presentation, visual aids and printed materials
- Be aware of your audience's dominant thinking mode and learning style (Herrmann Brain Model)
- Your audience has a busy schedule
  - Keep to your timeline
  - Practice your presentation and make sure that is designed for the time allowed
  - Manage questions to ensure you can cover your main points
  - Make sure you have enough time for your conclusion and summary

#### ❖ Written Technical Report

Technical reports are not essays or stories; the report is intended to communicate key information in an easy to understand and search format. Reports must be clear as to the problem being solved, research, analysis, recommendation and outstanding issues. An effective engineering report should contain the following sections, with clear titles and separation for each section:

- Cover page
  - Report title
  - Author
  - Date
  - A brief (1-2 paragraphs) executive summary (Abstract) of the problem and recommendation only.
- Problem definition
  - Statement of problem
  - Scope
  - Schedule
  - Resource
- Research
  - Publication research
  - Web research
  - Experimentation
  - Experts in the field

- **Analysis**  
Application of the research results in solving the problem.
- **Recommendation / Conclusion**  
The optimal decisions, implementations or solutions that are being recommended. It is important that the reader reaches similar conclusions upon reading earlier sections.
- **Issues / Dependencies**  
Issues or dependencies that have been encountered but have not been resolved as part of the work being reported.

## 2.6. Additional Resources

- ❖ Eide, A. Engineering Fundamentals & Problem Solving. (2002) McGraw Hill
- ❖ MathWorks. MATLAB Reference Material Version R2000a. (2007) MathWorks
- ❖ Spiegel, I. & Torres, C. Manager's Official Guide to Team Working (1994) Wiley

## 2.7. Problems

- 
1. Complete a self evaluation using the Herrmann Brain Model evaluation form. Include the resulting 4-category scores with a description of what each score implies and the radar chart.

---

  2. What are the four stages of Team Formations? Which do you believe is the most productive stage?

---

  3. What is your latest experience with negotiation? Explain the situation, and also identify the type of negotiation used. In retrospective, was it the best type of negotiation to be used for that situation?

---

  4. What are the top five considerations when presenting to a group of 30 people?

---

  5. Why does an engineer need to be skilled in communication and teamwork? Support your answer with examples.

---

  6. Identify an organization within 50 miles radius of your current residence that has the type of position that you are interested in applying for, either now or sometime in the future. Prepare a 5-minute presentation including visual aids. Your presentation should include:
    - Factors that resulted in you selecting the company
    - Key features of interest in the selected position
    - Your qualifications for this position and plans to acquire missing skills

---

  7. What are the four most important tools for managing teams?

---

  8. What are the negotiation types? Present a situation requiring negotiation and select the most appropriate negotiation type for it. Explain the reason you feel it is the most appropriate negotiation type?

---

  9. What are the four most important sections of written technical reports?

## **Chapter 3. Creative Problem Solving**

### **3.1. Key Concepts and Overview**

Step 1. Customer Issues/Needs Identification

Step 2. Problem Definition

Step 3. Idea Generation

Step 4. Optimal Solution Selection

Step 5. Solution Implementation

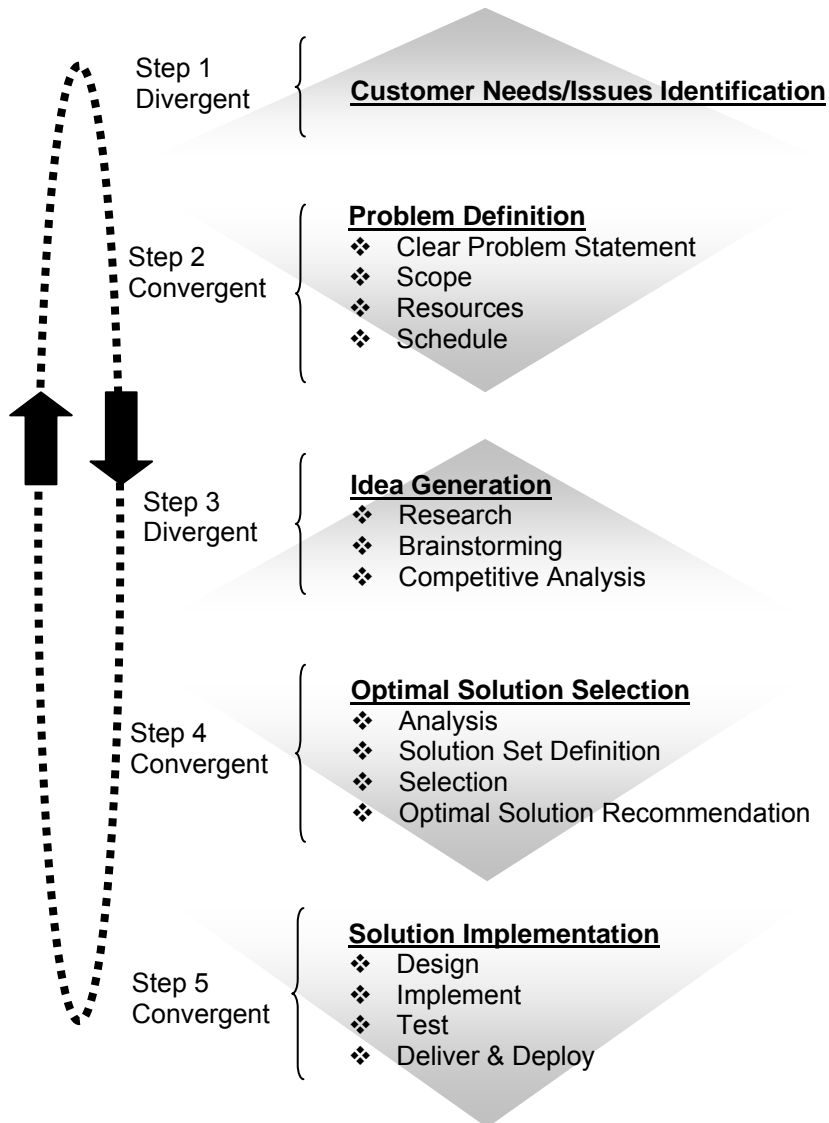
### **3.2. Creative Problem-Solving Process**

Creative problem-solving is the essence of engineering jobs. Creative problem-solving can be described in 5 steps:

- Step 1. Customer Issues/Needs Identification
- Step 2. Problem Definition
- Step 3. Idea Generation
- Step 4. Optimal Solution Selection
- Step 5. Solution Implementation

Although steps are shown consecutively, Creative Problem Solving is an iterative process; that is, results from latter steps may impact earlier steps in the process. In these situations, the earlier steps must be modified and all latter steps must be reworked.

## 5-STEP CREATIVE PROBLEM SOLVING PROCESS



### **Notes**

- 1) The Creative Problem Solving process is iterative with many feedback loops.
- 2) Convergent Thinking refers to the process of reducing the number of possibilities through a selection process.
- 3) Divergent Thinking refers to a process that expands the number of possibilities through a broadening of the scope and removing limitations.



### 3.3. Step 1 - Customer Issues/Needs Identification

Successful companies and engineers solve problems that add values for the customer. So in all cases, there exists one or more customers who either have unmet needs or issues that need to be addressed.

Typically, engineers work with marketing specialists and others to collect relevant information and compile them into a list which will be used to identify the problem.

The process to document customer issues and needs may include one or more of the following:

- ❖ **Personal experience**  
Engineers or people in their organization are experiencing the same problem, so they have first-hand experience and understanding of the problem.
  
- ❖ **Customer surveys and reports**  
Customer surveys and reports can provide a listing of unmet needs or issues.
  
- ❖ **Customer Support**  
The Customer Support team has identified a common issue or need with current product or service that should be addressed.
  
- ❖ **Competitive Analysis**  
Competitive analysis is the process that helps identify the relative strengths and weaknesses with respect to competitors. This information may be used to define the problem.
  
- ❖ **Trends**  
Market and/or technology trends among forecasting of future issues, unmet needs or opportunities.

Commonly, there are multiple sources, and the list may include hundreds of items.

**Student Exercise** - As a team, identify issues/needs that you believe if resolved, would add the highest value to your organization and/or community.

### 3.4. Step2 - Problem Definition

This is the most important step of creative problem solving, since solving the right problem only adds value. Regardless of the excellence of the final solution, if one is not addressing the right problem, then it is a waste of time and resources.

A complete problem definition has four distinct components:

- Problem Statement
- Scope
- Resources
- Schedule

#### ❖ Problem Definition Process

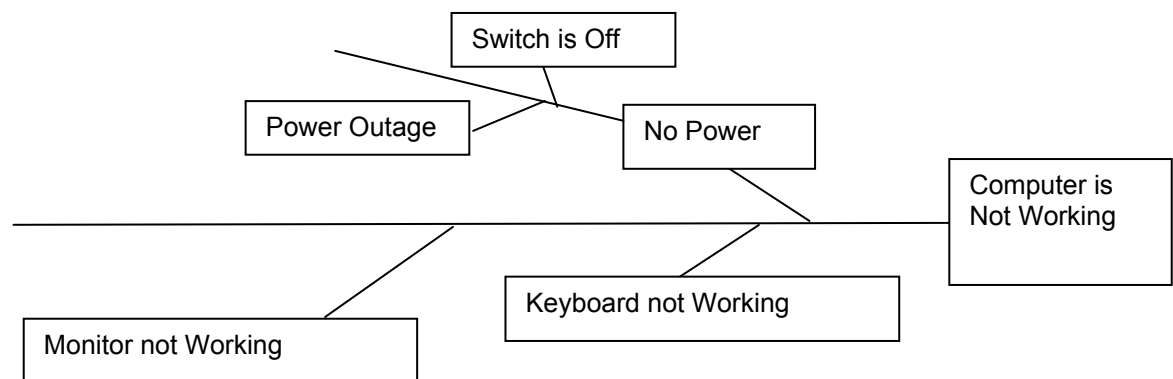
- Agree that there is a problem worth solving.
- Look for the root cause, not just the observable symptoms.
- Look at the trends in order to understand the problem's magnitude. Studying the trends also helps in understanding the development of problems in a wider context and timeframe.
- Research information in the problem areas. The goal is to learn from others who may have similar problems.
- Look at the problem in the context of its surroundings, and past and future trends. This approach is also called System Thinking where the problem is defined for the whole system not just one of part the system.
- Write the Problem Definition in term of its four components:
  - Problem Statement
  - Scope
  - Resources
  - Schedule

#### ❖ Methods to find the root cause

Understanding and solving root cause is important, since solving a symptom at best is a temporary solution and at worst is no solution. This Section describe the most common tools for identifying root cause:

- Ask questions (Kepner-Tregoe "KT" approach)
  - Who, What, Where, Why, how much, etc. questions?
  - It is a good technique for finding the problem boundaries
  - KT approach also identifies requirements that are outside of the scope of your problem and should not be included in the solution considerations.
- Survey  
Manufacturers and services companies use surveys to collect customer experience data. You may see this type of survey being called the "Voice of Customer", or other descriptive names. This data is usually presented in a Pareto chart (listing each problem and number of occurrences) which helps in deciding what problem to work on.

- **Statistical Process Control (SPC)**  
The SPC sets a limit for each parameter. When one parameter is out of the set range, then that item is identified as a problem to be worked on.
- **Failure Mode and Effects Analysis (FMEA)**  
Explore all possible failure modes of the product or process, and their effects on the result (customer experience).
- **Failure Tree Analysis (FTA)**  
Graphical view of the possible element failures that would result on a specific system failure.
- **Fishbone Diagram**  
Use a cause/effect diagram leading to the problem. Typically start with the problem as the head of the fish, and the possible causes of the problems are drawn as bones.



- **Experiments and Weibull Analysis**  
Carefully constructed experiments to answer a specific list of questions or to collect a specific set of data.
- **Benchmarking**  
Typically, use a world class competitor as the bench mark to see if your product or process has a comparative problem.
- **Introspection**  
When time is too short to do an in-depth data collection and analysis, one can think about what one knows and make a decision. Some say we do not spend enough time in this mode, while spending too much time asking others or collecting data that is never analyzed.

### 3.5 - Step 3 - Idea Generation

In this Section, the objective is to come up with as many ideas to solve the problem as is possible. The two steps in this process are:

- Research
- Brainstorming

It is important to understand the current state of knowledge with the respect to the problem being solved. So the first step is to access all sources and to compile all the relevant material:

Some common sources of information include:

- Networking - take advantage of meeting others and learning from their experiences.
- Searching the web
- Keeping an idea file
- Modeling a problem
- Searching relevant patents
- Experimenting
- Studying the Competitive solutions

Once all team members have an understanding of the problem and knowledge of existing relevant research data, then it is time for brainstorming possible solution ideas in a brainstorming process.

#### ❖ Brainstorming Rules

Although brainstorming is a creative process with as few boundaries as possible, in order to maximize the creative ideas, it is still important to have rules for an effective brainstorming. The four rules of brainstorming are as follows:

- No criticism is allowed – defer judgment until later. This is the most important rule!
- Generate as many solutions as possible – quantity counts, so use few words.
- Wild ideas are welcome – be as creative as possible and do not worry about practicality of your ideas.
- Leveraging is encouraged – build on the ideas of others

Brainstorming may be verbal or written, and is outlined here:

#### ❖ The Verbal (classic) Brainstorming Method

The key to successful brainstorming session is to be prepared, follow a procedure and debrief. Here are a some points to consider:

- Preparation
  - Team member
    - (1) Select several members with quadrant V preferences
    - (2) Have a direct or indirect representation from stakeholders (customers, design, test,...)
    - (3) The best number of people is 3-10.
  - Location

- (1) People think more creatively in unfamiliar locations, so hold the brainstorming off-site.
- (2) People should be seated so they can see each other and can easily hear each other
- Scheduling
  - (1) Earlier in the day is better, since brainstorming is exhausting work.
- Materials
  - Have the necessary equipment available to stimulate creativity: easel, flip charts, markers, note cards, visual aids, and props.

➤ Procedure

- Briefing
  - Allow team members a few minutes for social interaction and for each person to comfortably stake out a personal space in the seating arrangement.
- Review the 4 rules of brainstorming.
- Explain the procedure.
- Do warm up exercises.
- Brainstorming
  - (1) Ensure that the problem statement is well understood
  - (2) Ask team members to start sharing ideas. Start by bringing out the obvious in order to get it out of their minds and move forward to more creative solutions.
- Close
  - Once the idea generation has slowed down and time is running out; give a 5-minute warning. Typically, great ideas come out in the last 5 minutes.
- Dismissal
  - Collect all notes and ideas for later evaluation. Encourage team members to send you any ideas that come to them after the meeting.

➤ Debriefing

The best way to improve your brainstorming process is to find out what worked well and what did not.

❖ Written Brainstorming Methods

Written brainstorming methods help in situations with large numbers of participants or shy participants. For success, it is important that the four rules of brainstorming are followed and that the ideas are written down quickly with no regard to correctness of format or content.

➤ Pin Card Method

The Pin Card Method is applied by seating the participants around a large table and having them write down ideas on note cards – one idea per card. The note cards are passed and each member is asked to add his/her improvement or related idea to the card.

Student Exercise - Apply the Pin Card Method to ideas on improving this class. Start with a sheet of paper for each row and pass it down and back. Each person on each pass must add one item to the list.

➤ Crawford Slip Writing Method

The Crawford Slip Writing is used for large group brainstorming. Once the problem is defined, and each participant is asked to write down 20-30 ideas on slips of paper. Each idea should be written on a different slip of paper which is then collected quickly.

➤ Anonymous Method

A method with minimal face-to-face interaction is the Anonymous Method. An idea is written on paper and submitted to others in the organization. Each person can modify the idea or add their own. The important feature of this method is that the author of the idea remains anonymous.

➤ Panel Method

The Panel Method is used for very large groups, and is typically conducted according to the following steps:

- 5-7 volunteers are chosen from the entire group to be on the panel.
- The problem is presented to the whole group.
- The panel verbally brainstorms for 20 minutes, posting their ideas on flip charts.
- The rest of the group is able to add their new ideas or improvements to the list after the panel is done.

➤ Electronic Brainstorming

The Electronic Brainstorming uses online forums, bulletin boards or other electronic community formats in order to capture brainstorming ideas. Electronic brainstorming is used for large groups who are widely distributed through time, geography or other dimensions.

### 3.6. Step 4 - Optimal Solution Selection

As the name implies, this Section discusses the process to select the optimal solution. The Optimal Solution Selection is accomplished in the 2 steps shown below:

- Analysis and Solution Set Definition
- Optimal Solution Identification

Similar to Brainstorming, Optimal Solution Selection also has four rules:

- Look for quality and better ideas
- Make “wild” ideas more practical
- Synthesize ideas to obtain more complete, optimized solutions
- Maintain a positive attitude; continue to look for improving solution ideas

The first step of the process is Analysis and Solution Set Definition, which can be accomplished in the following three steps:

- Step 1: Sorting related ideas into categories
  - Lay out the ideas from the brainstorming on post-it notes or index cards. One idea per card or post-it note.
  - Allow team members to look at the cards for a period of time.
  - Ask them to put cards with similar ideas together.
  - Add category title cards on each grouping of similar ideas as they form. Use either color cards or symbols that have nothing to do with the area of discussion; you do not want to create any presumption of category definition.
  - If an idea fits into more than one group then create a duplicate card.
  - If an idea does not fit in any existing category groups then create a new category.
  - You should keep the final number of categories to 5-7. If there are more, repeat the process to see if any can be combined.
- Step 2: Developing quality ideas within a category
  - Work on one category at a time. If you have a large team (more than 7), then create multiple teams and have each team work on a separate category at a different table or location.
  - Let each team know that the goal of this step is to “engineer” the many ideas within the category down to fewer, and more completely developed, practical and higher-quality ideas. This is called “Idea Synthesis”.
  - The combined idea is written on the same or a new card, but make sure to keep the original idea attached to the new synthesized idea. Follow the process until the category is completed.
  - Danger: The team gets focused on one novel idea and ignores the rest.
- Step 3: Force-fitting unrelated ideas between categories
  - This is where the whole team comes together and tries to combine ideas from all the categories in order to come up with a superior solution.
  - In some situations, it is acceptable to delay the process until a later date if the situation dictates it.

Once Analysis and Solution Set Definition has been completed, we will have 3-5 more practical and better-defined solutions without loss of any creativity. So now is the time to work on Optimal Solution Identification from this smaller set. At this point, we can use the following three steps in order to pick solutions to recommend and implement:

- 1) Establish evaluation criteria; list of solution attributes that are needed.

- 2) Rank ideas/solutions based on the evaluation criteria.
- 3) Decide on the best solution or idea to be implemented.

Now, let's discussed details involved with the each of the steps:

❖ **Step1. Selection Criteria**

A good list of selection criteria includes all factors that influence a problem or decision. In selecting the Selection Criteria, consider the following factors:

- **Balanced list** - analytical and intuitive criteria
- **Boundaries** - scope, schedule and resource limitations
- **Projection** - look to the future and consider factors that would simplify the implementation and make the sell and support process more effective among other factors.

❖ **Step 2. Ranking Ideas**

Rarely one idea emerges that will meet all of the selection criteria, so typically, there is a need to think of ways to rank possible solutions. The approach you pick will depend on the timeline, complexity and importance of the decision.

- **Pure Voting**
  - Each person votes based on his/her understanding of optimal solutions.
  - **Issue:** Any potential benefit that may come from discussion is lost and may also be negatively affected by peer-pressure/group thinking.
- **Voting Variations**
  - **Agreement gradient**  
Allow each person to vote from 1 (cannot live with the idea) to 10 (the best idea ever). This method enables the participants to discuss their choices.
  - **Ranking**  
Instead of picking one idea, each person ranks ideas in order of preference and discusses their choice. Many companies use this method for selecting employees.
- **Advantage/Disadvantage Technique**  
The most useful way is to add weighting factors to each criterion and then compare how each idea ranks vs. the criteria. Two common weighting approaches used are:
  - + for positive, - for negative and 0 for no impact
  - -10 for large negative impact and + 10 for large positive impact

Here is an example of a Selection Criteria Table used to document the ranking of different degrees using the -10 to +10, each criteria weighted (note you can get as detailed as you need to be):

<b>Criteria (Weight)</b>	<b>Ideas / Potential Solutions / Options</b>		
	<b>Comp. Sc.</b>	<b>Comp. Eng.</b>	<b>Elect. Eng.</b>
Criteria 1 (x20)			
Criteria 2 (x50)			
Criteria 3 (x30)			
...			

- “\$100 decision-making” is a variation of ranking process where you have \$100 and can spend it on the criteria according to importance.



- **Advocacy Method**  
The Advocacy Method works best for a small number of ideas. Each team member is assigned one idea to defend in front of the team. The team makes the final decision based on the presentations.
- **Reverse Brainstorming Method**  
In the Reverse Brainstorming Method, each member is assigned one idea to criticize its weaknesses and flaws. Finally, the team decides amongst the ideas.
- **Experimentation or Taguchi Method**  
The Experimentation, or Taguchi Method, relies on well designed experiments and trial-and-error methods in order to decide on the best solutions.

The best (extreme) example of this approach is the work on the light bulb. Thomas Edison and his lab used over 1,000 different materials to identify the material that would work best as a filament in a light bulb.

### ❖ Step 3. Decision Making

The ranking methods do not always come up with a clear method, so here are some common ways decisions are made:

- **Coin Toss**  
If ideas are equally good, then coin toss (random selection) may be the best answer.
- **Easy Way Out**  
If ideas are equal in merit, then implementation considerations may be used. Select the final solution based on the ease of implementations.
- **Refine Criteria**  
If the first set of criteria did not result in a selection of the solutions, this is to refine or add criteria to better separate various solutions. The additional or Refine Criteria may show advantages and disadvantages that might have been missed the first time through the selection process.
- **Consensus**  
If time is short, you may want to try for Consensus, which means select the idea that everyone can agree to. The result of Consensus process is typically poor, because participants go for the easy and obvious ideas, avoiding creative, new ideas.
- **Hybrid Decision**  
If there is sufficient time, a Hybrid Decision Method can be used. This method synthesizes the best solutions by integrating the best of each idea. This is the highest form of group decision-making, and it is a highly recommended approach for creative solutions.
- **Compromise**  
The Compromise Method reaches a decision through trade-offs among the top selected solutions. This form of decision-making is commonly used in government and is not recommended for creative problem solving.
- **No Decision or Delay**  
Sometime it is wise to delay and not make a decision just as long as it is recognized that no decision or delay is also a decision. If the decision is delayed to collect more information, and not because no one wants to take a risk, then it should be considered. In all other cases this would be problematic.
- **Intuitive**

For many people, intuitive decision making is very effective. It is common to come up with a decision intuitively and then rationalize it. Although this approach is not commonly accepted, it is important to give appropriate weight to intuition in final decision-making.

❖ Did you make the right decision?

Now that the decision is made, how can one be sure that the decision is correct?. This is a very important question that should be asked every time a decision is made. It can help find flaws in the decision and also help to improve future decision-making processes.

Here is a checklist of activities to consider:

- Can the idea be combined to obtain a higher-quality solution?
- Can different ideas of equal quality be implemented all at once?
- How well does the solution or idea solve the problem or meet the problem definition?
- Validate that the ideas meet all stated needs.
- Complete a risk analysis on implementing the decision.
- Conduct a cost/benefit analysis.

The only way to know the goodness of a decision is through the test of time and results. So it is important to review the results once they are known and use them to improve your decision making process. This is referred to as Continuous Process Improvement (CPI) and should be applied to all processes.

### 3.7. Step 5 - Solution Implementation

Implementation of the selected solution requires the following steps:

- Design
- Build
- Test
- Deliver
- Deploy
- Support

Although the process is shown as a linear set of steps, in practice, it is an iterative process. Each step may yield information that affects earlier phases. A good design requires that new findings be used to improve earlier steps if appropriate.

### **3.8. Additional Resources**

- ❖ Lumsdaine, E. Creative Problem Solving and Engineering. (1999) Prentice Hall.
- ❖ Eide, A. Engineering Fundamentals & Problem Solving. (2002) McGraw Hill

### 3.9. Problems

---

1. Present a problem and apply the five-step Creative Problem-Solving technique to it.

---
2. What is the most appropriate brainstorming process for a team with 40 members?

---
3. Use a Fishbone Diagram to perform a root cause analysis for a computer that does not turn on.  
*Note: Be prepared to present in class.*

---
4. What are the top three problems facing humanity at this time? Explain the reasons for your choice. What are the trends relating to the three problems selected?  
*Note: Compare your results with your teammates, and be prepared to present in class.*

---
5. Work with your team and as a group, identify a problem that can be addressed using computing/electrical related concepts. Apply the five-step Creative Problem Solving technique to the selected problem. Include the Thinking Model for each team member.

---
6. Name the steps in the Creative problem solving that are Convergent.

---
7. What are the four rules of Creative idea generation?

---
8. Describe the weighted advantage and Disadvantage ranking technique?

---
9. Describe the Anonymous method and the advantage of this method over Pin Card method.

---
10. Name the stages of team development. Describe the status of your lab team with respect to team development stages.

---
11. Name the quadrants in Herrmann Brain Domain Model (HBDM) and state the overriding communication need for each quadrant.

---

## **Chapter 4. Electrical Circuits**

### **4.1. Key Concepts and Overview**

- ❖ Charge, Current and Voltage
- ❖ Circuit Model
- ❖ Power Sources
- ❖ Kirchoff's Laws and DC Circuit Analysis
- ❖ Circuit Simplification
- ❖ Capacitors, C
- ❖ Inductors, L
- ❖ Transformers

## 4.2. Charge, Current and Voltage

### ❖ Electric Charge

Charge in physics may be electric, color or magnetic. We will be focusing on electric charge here. Electric charge is the basic property of some subatomic particles. Electrons by convention have a charge of -1 and similarly the protons have the opposite Charge of +1. In general, electric charge is bipolar, meaning that electrical effects are described as having positive and negative charges.

The SI unit for electric charge is the Coulomb, which represents approximately  $6.24 \times 10^{18}$  elementary charges. Elementary Charge is the charge on a single electron or proton. In other word electric charge exists in discrete quantities (elementary charge) which are integral multiples of the electric charge,  $1.6022 \times 10^{-19}$  C.

### ❖ Electric Current and Voltage

Electrical effects are attributed to both the separation of charge (Voltage) and charges in motion (Current).

#### ➤ Current

Amount of current is measured in amperes and at its simplest form is the change of charge q over

times which may be expressed as  $i = \frac{\Delta q}{\Delta t}$  where:

i = current in Amperes  
q =charge in Coulombs  
t = time in Seconds

#### ➤ Voltage

Amount of voltage is measured in Volts and it represent the amount of energy required to move

electric charge from point a to b. Therefore voltage is expressed as  $v = \frac{\Delta e}{\Delta q}$  where:

v = voltage in Volts  
e = energy in Joules  
q = charge in Coulombs

Voltage, V, is the difference of voltage in two points in the circuits.

### ❖ Analogy Between Electricity and Water

Since we all have personal sensory experience with water flow and pressure, it may be useful to compare attributes of water flow through a pipe with electricity through wire (conductor). Below is a comparison of electricity and water key attributes:

WATER	ELECTRICITY
Pressure	Voltage
Flow Rate	Current
# of Gallons	Charge
Resistance of Pipe	Resistance of wire

❖ Powers of 10

Engineers typically use powers of 10 that are multiple of three in order to streamline communications of very small and very large numbers. Below are list of the most commonly used powers of 10 in engineering:

➤ Factors more than 1

- $10^{24}$  yotta (Y)
- $10^{21}$  zetta (Z)
- $10^{18}$  exa (E)
- $10^{15}$  peta (P)
- $10^{12}$  tera (T)
- $10^9$  giga (G)
- $10^6$  mega (M)
- $10^3$  kilo (k)

➤ Factors less than 1

- $10^{-3}$  milli (m)
- $10^{-6}$  micro ( $\mu$ )
- $10^{-9}$  nano (n)
- $10^{-12}$  pico (p)
- $10^{-15}$  femto (f)
- $10^{-18}$  atto (a)
- $10^{-21}$  zepto (z)
- $10^{-24}$  yocto (y)



### 4.3. Circuit Model

An Electrical Circuit is a mathematical model that approximates the behavior of an actual electrical system. Ideal Circuit theory is a special case of electromagnetic field theory. The three assumptions that allow the use of the ideal Circuit theory are:

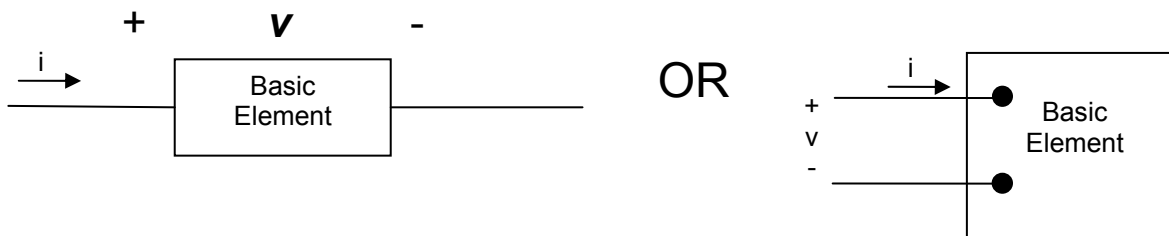
- Electrical effects propagate instantly through the whole system  
The systems that are physically small relative to signal wavelength for this assumption to be valid are called lumped systems. If the wavelength of the signal is more than 10 times the physical dimension of the system then we have a lumped system. As a reminder, the formula for calculating of wavelength is shown below:

$$\text{Wavelength } \lambda = (\text{speed of light, } 3 \times 10^8 \text{ m/s}) / (\text{signal frequency, } f \text{ Hertz})$$

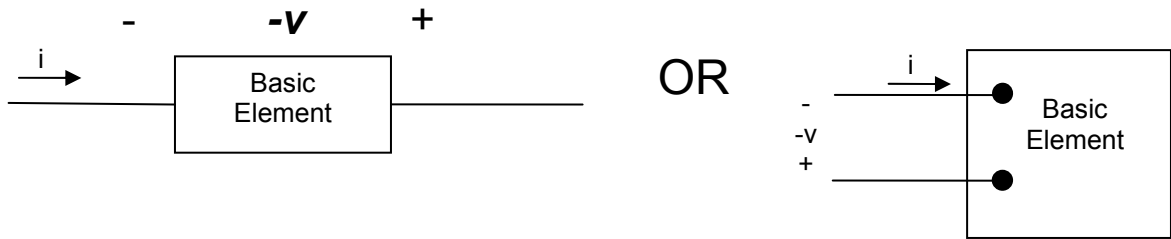
Student Exercise: If signal frequency is 20 MHz, what is the maximum size for a circuit that can be classified as a lumped circuit.

- Net charge of every component in the system is zero  
This assumption states that none of the components collect net charges or lose net charges. In other word, the change that enters a component, must leave the component.
  - No magnetic couplings  
No magnetic coupling means that charges can only flow through physically connected components by a conducting material (typically wires). It follows that with this assumption, charges cannot leave the circuit through air.
- ❖ Circuit analysis uses idealized basic circuit elements and ideal circuit model (a mathematical model) in this text. Earlier the characteristic of ideal circuit model was defined and the following three attributes characterize an idealized basic circuit element:
- It has only two terminals which are the points it connects to other elements.
  - It can be fully described in term of relationship between current and voltage.
  - It cannot be subdivided into other elements.

Electrical Engineering typically uses the passive convention which assumes the current enters at positive voltage terminal. Two layout of a basic element is shown below using passive convention:



You may use negative sign to reverse the voltage polarity or reverse current direction. For example the above model may be redrawn as:



❖ Ideal Basic Elements

The rest of this text uses the following four basic ideal elements to model electrical system. As we progress through the material, one-by-one the simplifying assumptions will be removed to better approximate the actual electrical characteristic of the system.

The Four Elements are listed below:

- Power source
- Resistor
- Capacitor
- Inductor

Each of these elements will be defined in latter chapters.

## 4.4. Power Sources

The first step in analyzing any system is to learn about the components and rules that govern their operations in the system. The first component is power sources which provides the energy required for the operation of the circuit. Our focus here is on Direct Current (DC) circuit which means the power sources provide either constant voltage, which is called the ideal voltage source, or constant current, which is called ideal current source. The word “ideal” in the title means that the source does not have any internal inefficiencies. In more practical terms, the ideal source is able to produce the specific level of current or voltage regardless of the load. Of course, we know that such a source does not truly exist, but this simplification helps us get started with analysis and in future studies, the imperfections will be added to the description.

### ❖ Power Sources

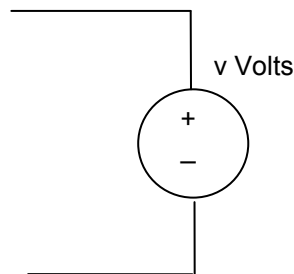
This section outlines the two types of power sources and discusses their characteristics.

#### ➤ Independent Ideal Voltage Source

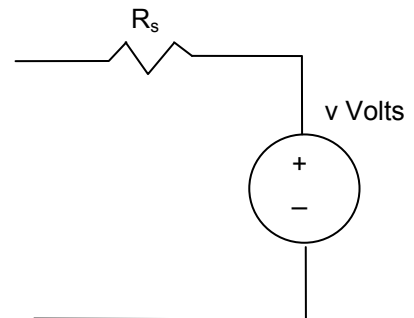
An Independent Ideal Voltage Sources, as the name implies, has the following characteristics:

- Supplies designated voltage regardless of the rest of circuit.
- It does not consume any of the power internally.

The following diagrams show the symbol for Independent Ideal Voltage source (a) and Independent non-ideal Voltage Source (b). The difference between ideal and non-ideal is that non-ideal has an internal resistance which consumes some of the power internally which will not be available for the rest of the circuit to use.



a. Independent Ideal Voltage Source



b. Independent non-ideal Voltage Source

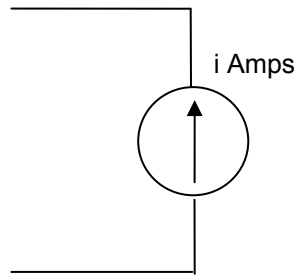
In the remainder of this text ideal source will be used.

#### ➤ Independent Ideal Current Source

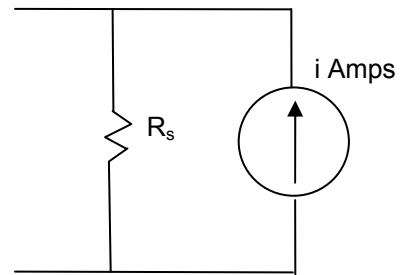
An Independent Ideal Current Source, as the name implies, has the following characteristics:

- Supplies designated current regardless of the rest of circuit.
- It does not consume any of the power internally.

The following diagrams show the symbol for Independent Ideal Current Source (a) and Independent non-ideal Current Source (b). The difference between ideal and non-ideal is that non-ideal has an internal resistance which consumes some of the power internally which will not be available for the rest of the circuit to use.



a. Independent Ideal Current Source



b. Independent non-ideal Current Source

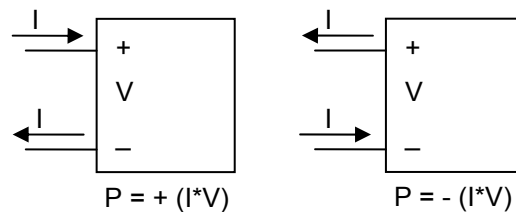
❖ Power Calculation

Power is the key parameter in Electrical Engineering since the signal's power over time translates to the work done, and also the signal's power contains information. Power ( $P$ ) measured in watts is a product of Current ( $I$ ) and Voltage ( $V$ ).

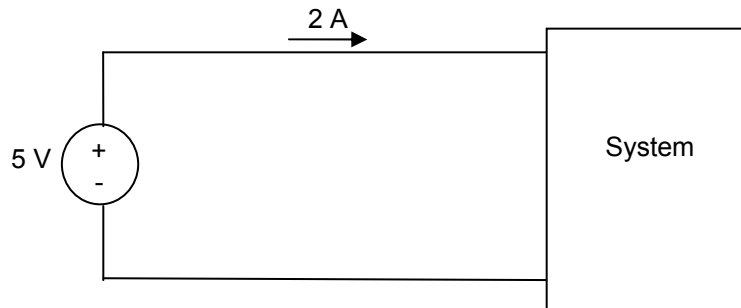
$$P = V \cdot I \quad \text{when } P \text{ is in Watts, } V \text{ is in Volts and } I \text{ is in Amps.}$$

An electrical element may generate or consume power. "Passive Convention" is used in Electrical Engineering which states that positive power value indicates that the element is consuming power and negative power value indicates the element is producing power.

In "Passive Convention", when current flows into the positive voltage terminal of an element, the power is "+". The following diagram depicts all possible configurations and corresponding power signs.



➤ Example - Find the Power at the voltage source and at the system:

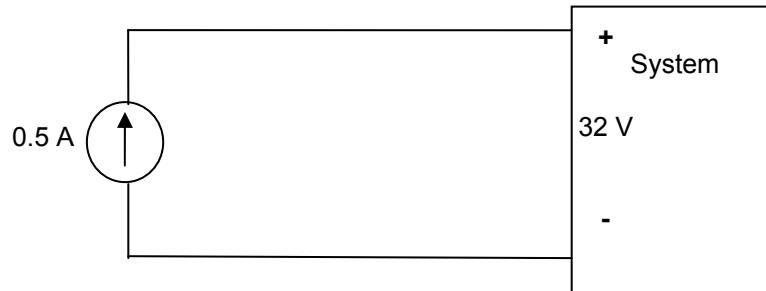


Solution

$$P_{\text{Voltage Source}} = -(I \cdot V) = -(2 \cdot 5) = -10 \text{ W} \quad \text{"Generating Power"}$$

$$P_{\text{System}} = (I \cdot V) = (2 \cdot 5) = 10 \text{ W} \quad \text{"Consuming Power"}$$

- Student Exercise - Find the Power at the current source and the system for the following circuit. Also, specify if the power is being consumed or generated.:

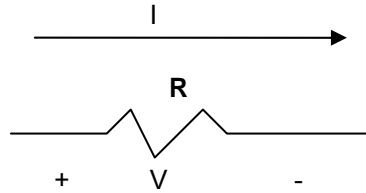


### Solution

- ❖ Properties of the Ideal Circuit in Terms of Current (I), Voltage (V) and Power (P)  
As discussed earlier, an ideal circuit is a mathematical model. A valid circuit must meet all of the following three conditions:
  - Current measured on various parts of a connecting line are the same.
  - Voltage across two terminals is the same regardless of which side of the terminal measurement is made.
  - The sum of the element powers in a circuit equals zero (Developed Power = Absorbed Power).
- ❖ Passive vs. Active Elements  
One way to classify components is based on their ability to generate power. If the element can generate power, then it is classified as active; otherwise, the element is passive. We have five major circuit components that are used to model all electrical and electronic systems:
  - Active: Current and Voltage Sources  
The value of power is negative for active devices since they generate power.
  - Passive: Resistors, Capacitors and Inductors  
The value of power is positive for passive devices, since they consume power.
- ❖ Electrical Resistance  
Resistance is the capacity of a given material to impede the flow of current (electric charge). The flow of current through a resistor will convert electrical energy to thermal energy. In some applications, this property is desirable and in other applications, it is undesirable. Here are examples of each:
  - Undesirable: transmission line, digital devices
  - Desirable: heater, toaster, oven, cook top

In circuits, the resistance of material is modeled as resistors and is measured in Ohms.

➤ Resistor Symbol



➤ Ohm's Law

Resistance is a basic ideal element and its value is defined in terms of current,  $I$ , and voltage,  $V$ . George Simon Ohm, a German physicist, came up with Ohm's Law in 1827. Although Ohm's law is the basis of everything electrical, his colleagues at the time did not believe the law and caused Ohm to lose his teaching position. Being an innovator is tough but has long-term benefits!

Ohm's Law is a fundamental relationship between voltage, current and resistance as shown below:

$V = I * R$  where:

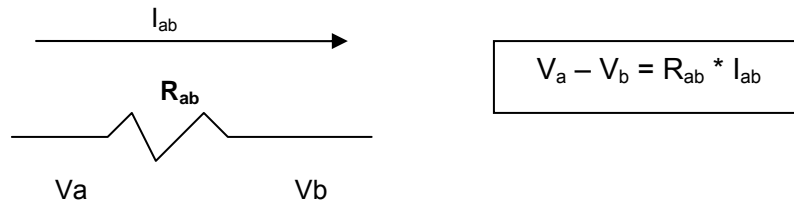
- \* R value is in Ohms or  $\Omega$
- \* V value is in volts
- \* I value is in Amperes

Ohm's Law enables us to use algebra to find any of the three parameters when the other two are given. The three forms of Ohm's Law are shown below:

$V = I * R$       or       $R = V/I$       or       $I = V/R$

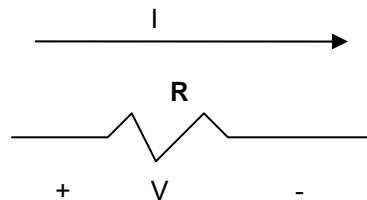
A more detailed statement of Ohm's Law:

The voltage difference between two points is equal to the current between the points times the resistance between the points, or



Passive Sign Convention

We will use the passive sign convention, which requires that current flows from the positive terminal to the negative voltage terminal for passive devices such as a resistor.



Conductance (G) is a measure of how well electricity is conducted and it is inversely proportional to resistance.

$$G = 1/R$$

Conductance is formally measured in Siemens (S) but it is commonly referred to as Mhos.

❖ Resistor Power Definitions

Power is the amount of energy transferred in one second. Power as a proxy for energy is a critical factor since it dictates how bright a light is, how fast a motor is rotating, and even the information contained in an electrical signal.

Starting with the fundamental power, current voltage relationship shown below:

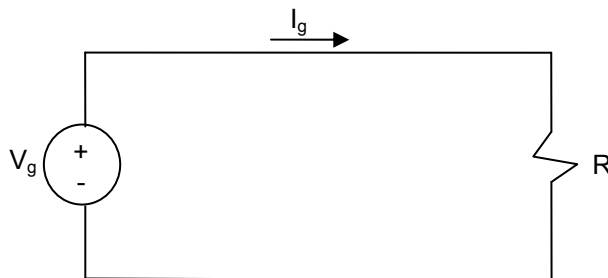
- $P = V * I$  where  
P is in Watts,  
V is in Volts,  
I is in Amps.  
(note: I is positive if it flows to the positive terminal and negative if it flows to the negative terminal.)

We can rewrite the power relationship in terms of R and either V or I by simple substitution as shown below:

- $P = V^2 / R$  (I is replaced using the equation  $I=V/R$ )
- $P = I^2 * R$  (V is replaced using the equation  $V=IR$ )

❖ Examples – Application of Ideal Circuit, Resistor, Power Sources, Ohm’s Law and Power.

- Example - Find the value of R and the power consumed by the resistor if  $V_g = 1$  KV and  $I_g = 5$  mA.

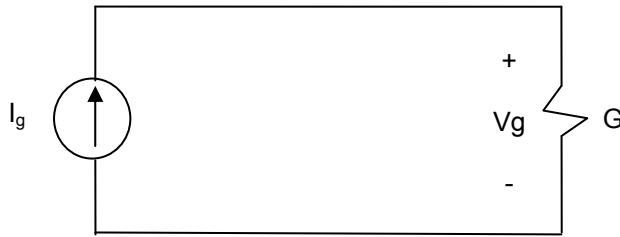


Solution

$$R = V_g / I_g = 1000 / 0.005 = 200 \text{ k}\Omega$$

$$P_r = I_g^2 * R = (0.005)^2 * (200,000) = 5 \text{ W}$$

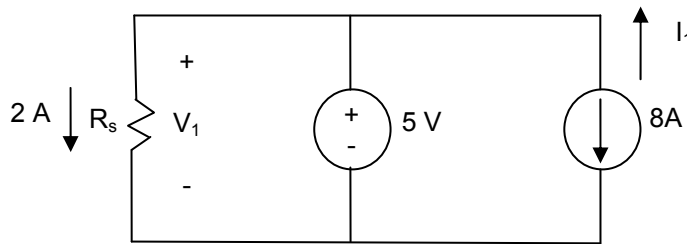
- Example - If  $I_g = 0.5$  A and  $G = 50$  m Mhos, find  $V_g$  and the power delivered by the current source. Note: Conductance,  $G = 1/R$  Mhos, is a measure of how well the element conducts.



Solution

- $V_g = I_g / G = 0.5 / 0.05 = 10 \text{ V}$   
 $P_G = I_g^2 / G = (.5)^2 / (0.05) = 5 \text{ W}$

➤ Example - Determine the current  $I_1$ ,  $V_1$  and  $R_s$  for the following valid circuit:



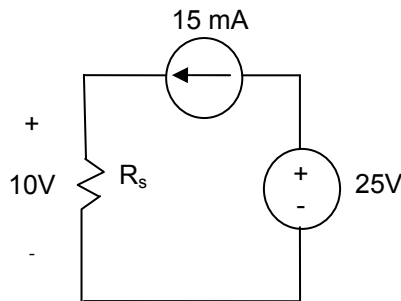
Solution

$I_1 = -8 \text{ A}$

$V_1 = 5 \text{ V}$

$R_s = V/I = 5 / 2 = 2.5 \Omega$

➤ Example – Calculate the value of  $R_s$  and determine if the following the circuit is valid ideal circuit:



Solution

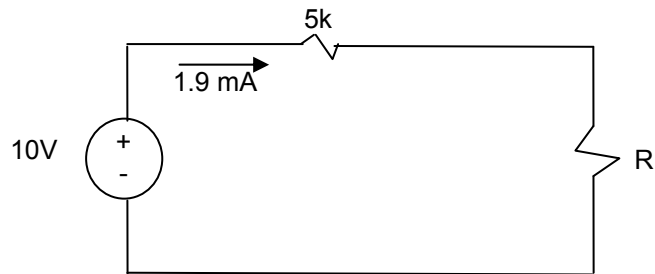
In an ideal circuit current, at any part of a branch is the same  $\rightarrow R_s = V/I = 10 / 0.015 = 667 \Omega$

In a valid circuit  $\sum \text{power} = 0$

$\sum \text{power} = P_{\text{resistor}} + P_{\text{current source}} + P_{\text{voltage source}} = 10 * 0.015 - 25 * 0.015 + (25 - 10) 0.015 = 0$   
 Therefore Circuit is valid.



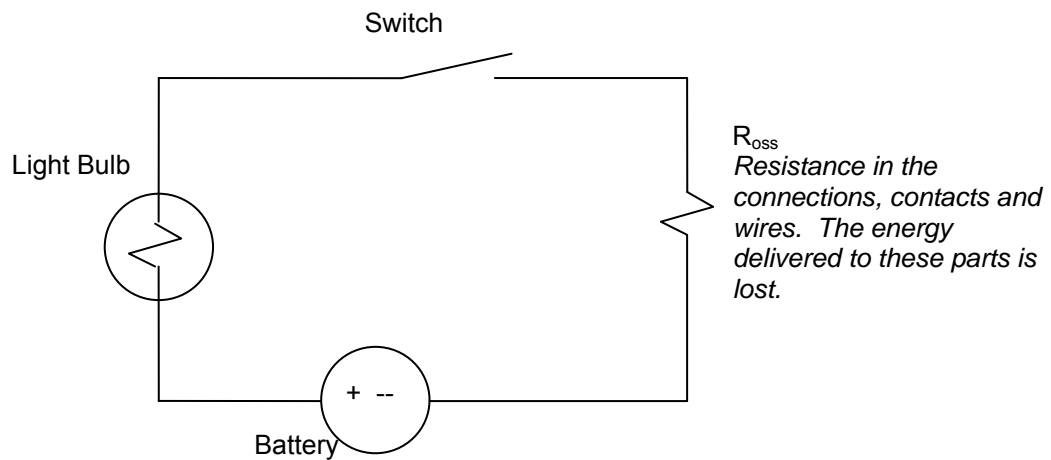
- **Student Exercise** – Find the value of R in the following circuit and determine if the following circuit is valid ideal circuit:



Solution

- Example - Construct a circuit model for a flash light using electrical components.

Solution



- A light bulb is the resistor that converts electrical energy to high temperature that causes radiation in the visible range

- A battery is the power source. It is not an ideal source, since the power drops over the life of the battery.
- Switch
- Considerations
  - (1) Heat generated
  - (2) Power requirement
  - (3) Parasitic effects (connections, switches, coils)

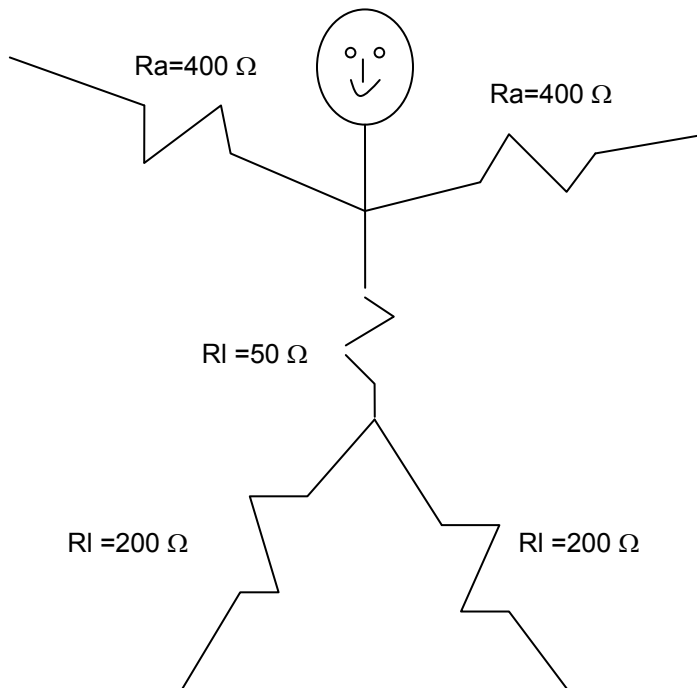
➤ **Student Exercise** – Answer the following questions for a flash light that has two D-cell batteries in series (each D-Cell Battery has 1.5 V and 2.5 Amp Hour), a bulb with 5 ohms of resistance and interconnection resistance of 1 ohm:

- a) What is the current through the light bulb?
- b) What is the power generated by the two batteries?
- c) How long would the flash light last if it used continuously under ideal condition?

Solution

- Example - Construct an electrical circuit model for the human body

Solution



Although electricity burns due to high voltage is a problem, it rarely causes death.

The effect of electrical current on human nervous system is very severe, especially if the muscles being controlled are of the heart.

Here are various current levels and the corresponding physiological responses:

<u>Current</u>	<u>Physiological Responses</u>
3-5 mA	Barely Perceptible
35-50 mA	Extreme Pain
50 – 70 mA	Muscle Paralysis
> 500 mA	Heart Stoppage

- **Student Exercise** - What is the physiological impact of 220 V being shorted between the person's left hand and right leg?

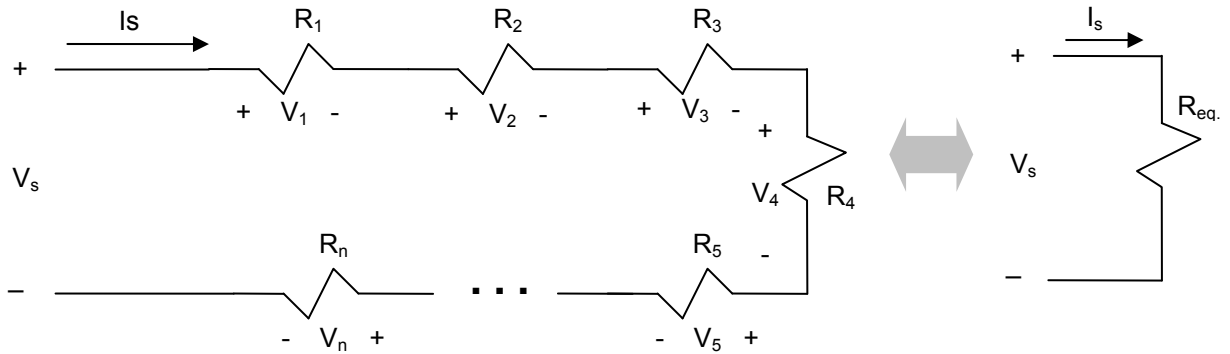
Solution

## 4.5. Circuit Simplifications

For a significant number of circuits, analysis may be simplified by replacing multiple resistors with a single equivalent resistor. This section discusses two types of resistor networks that may be replaced by an equivalent resistor are series resistors and parallel resistors networks.

### ❖ Resistors in Series

When two elements connect at a single node and have the same current flow, they are said to be in a series. The following circuit is a generalized form of  $n$  resistors in a series:



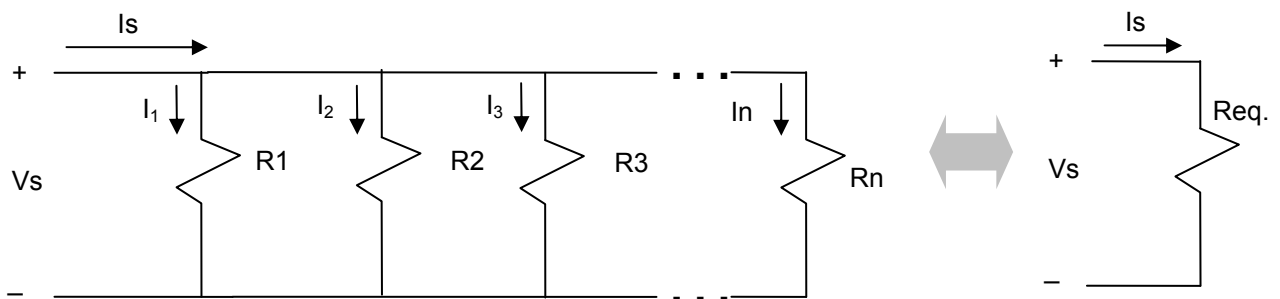
The equivalent resistor for a series resistor network can be calculated using the following equation:

$$V_s = V_1 + V_2 + \dots + V_n = I_s R_1 + I_s R_2 + \dots + I_s R_n = I_s R_{eq}$$

$$R_{eq} = R_1 + R_2 + \dots + R_n = \sum_{i=1}^n R_i \quad \text{for series resistors}$$

### ❖ Resistors in Parallel

Two resistors are said to be connected in parallel when terminals of resistors are connected together so both resistors have the same voltage across them. The following circuit is a generalized form of  $n$  resistors in parallel:



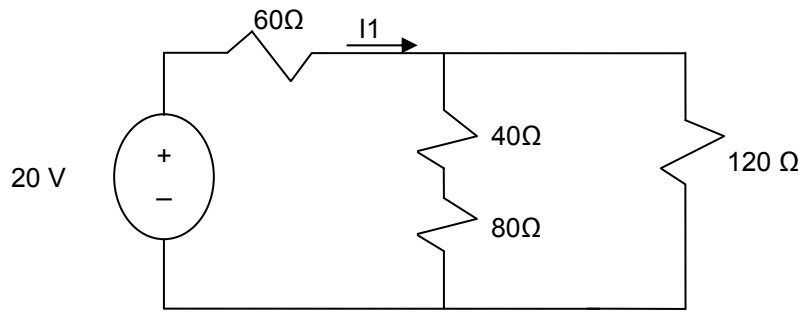
The equivalent resistor for a parallel resistor network can be calculated using the following equation:

$$I_s = I_1 + I_2 + \dots + I_n = V_s/R_1 + V_s/R_2 + \dots + V_s/R_n = V_s/R_{eq}$$

$$1/R_{eq} = 1/R_1 + 1/R_2 + 1/R_3 + \dots + 1/R_n = \sum_{i=1}^n \frac{1}{R_i} \quad \text{for parallel resistors}$$

### ❖ Example – Circuit Simplification

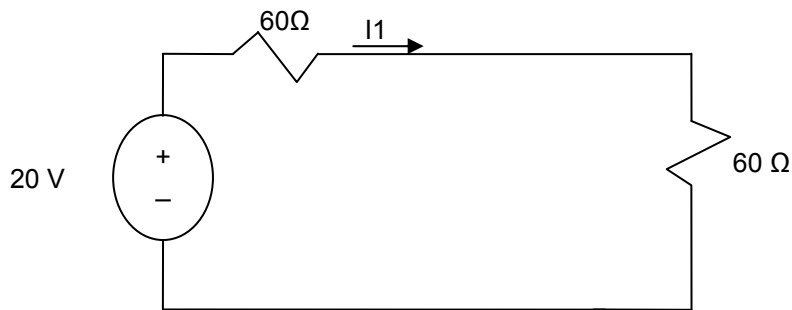
For the following circuit determine the value of  $I_1$  using the resistor simplifications.



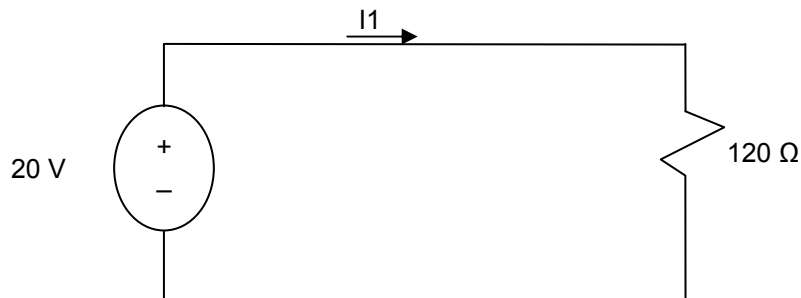
Solution

- \* 40Ω and 80Ω are in series, therefore they can be replaced with  $R_1 = 40 + 80 = 120 \Omega$
- \*  $R_1=120 \Omega$  and  $120 \Omega$  are in parallel, therefore  $1/R_1 = 1/120 + 1/120 \rightarrow R_2 = 60 \Omega$

Therefore, the circuit can be redrawn as:

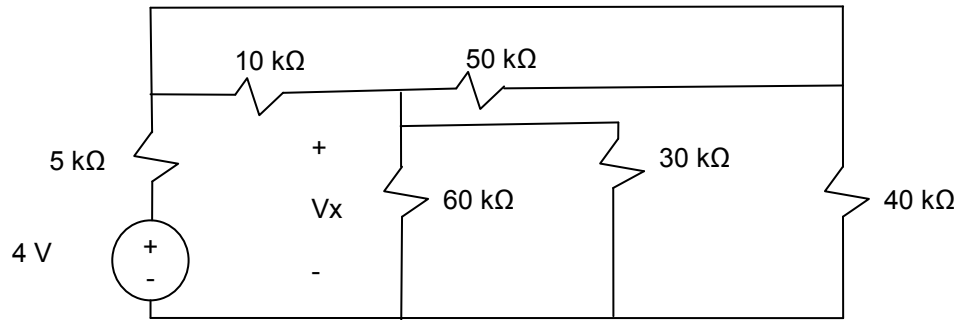


- \* The two 60 Ω resistors are in series, therefore they can be replaced with  $R_3 = 60 + 60 = 120 \Omega$



$$I_1 = V/R_3 = 20/120 = 1/6 \text{ A}$$

- ❖ Student Exercise – Use Resistor simplification to find the current flow through 4 V sources and the power generated by the 4 V source.



Solution

## 4.6. Kirchhoff's Laws and DC Circuit Analysis

Solving a circuit problem or analyzing circuits mean finding currents and voltages in the circuit. Kirchhoff's Laws are useful tools in circuit analysis. There are two Kirchhoff's Laws:

- Kirchhoff's Current Law (KCL)
- Kirchhoff's Voltage Law (KVL)

### ❖ Kirchhoff's Current Law (KCL)

Kirchhoff's Current Law (KCL) states that "The algebraic sum of all the currents at any (essential) node in a given circuit equals zero" where:

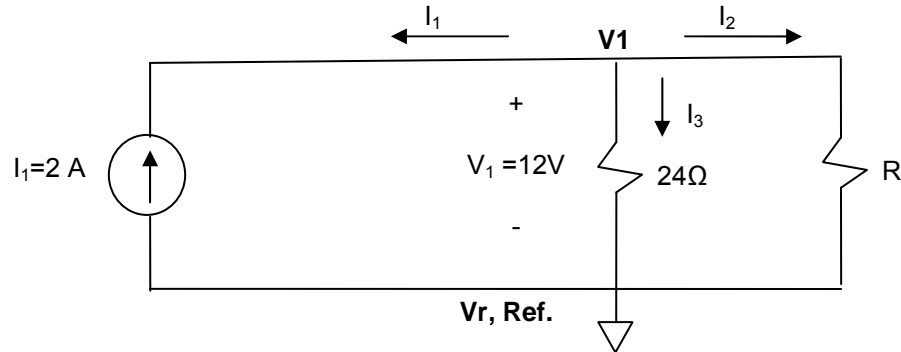
- An (essential) node is a point where MORE than two elements meet.
- In this text, the current going out of the selected node is considered positive, which means the current flowing into the node is considered negative.
- In a circuit with "n" nodes, there are "n-1" independent current equations, since we will assume one node as reference ( $V=0$ ).

The following four steps are a simplified approach to performing KCL analysis:

- (1) Identify the nodes
  - (a) Pick one node as reference ( $V_r=0$ )
  - (b) Label node voltages:  $V_1, V_2, \dots$   
*Note that these are voltage differences between the designated node and reference node.*
- (2) Draw current arrows
  - (a) Draw all current arrows away from the nodes (Positive).  
*Remember the current direction is positive going out of the node.*
  - (b) Label currents:  $I_1, I_2, \dots$
- (3) Apply KCL - write the KCL equation for each node  
Sum of currents leaving each node equals zero.
- (4) Apply Ohm's Law ( $I=V/R$ ) to re-write the KCL equations in term of voltages.  
The objective here is to reduce the number of unknowns in the equations.:

The next example is intended to apply the KCL in a step-by-step approach to analyze circuits. It is important to be consistent in applying of KCL in order to improve accuracy and speed of analysis.

- Example - Kirchhoff's Current Law Application  
Apply Kirchhoff's Current Law to find the value of R.



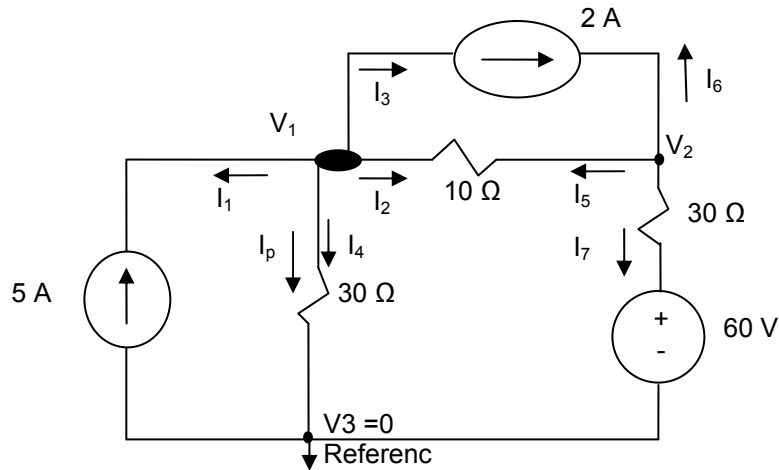
Solution:

- (1) Identify the nodes
  - (a) Pick one node as reference ( $V_r=0$ )
  - (b) Label node voltages:  $V_1$   
Note that these are voltage differences between the designated node and reference node.
- (2) Draw current arrows
  - (a) Draw all current arrows away from the nodes (Positive). Remember the current direction is positive going out of the node.
  - (b) Label currents:  $I_1, I_2, I_3$ .
- (3) Apply KCL - write the equation for node  $V_1$   
equation  $V_1$  (outgoing current is positive)  $\rightarrow I_1 + I_2 + I_3 = 0$
- (4) Apply Ohm's Law ( $I=V/R$ ) to re-write the KCL equation(s) in term of voltages (reduce number of unknowns):
  - (a)  $I_1$  is given as  $-2A$  (Opposite the current source)
  - (b)  $I_3 = V/R = (12/24) = 0.5 A$  (Ohms Law)
  - (c) Apply values to KCL equation  
 $I_1 + I_2 + I_3 = 0 \rightarrow -2 + I_2 + 0.5 = 0 \rightarrow I_2 = 1.5$

$$\text{We know that } V_1 = I_2 * R \rightarrow R = 12/1.5 = 8 \Omega$$



❖ Example - Find  $I_p$  in the following circuit using KCL:



Solution:

(1) Find all the essential nodes.

There are three nodes:  $V_1, V_2, V_3$   
Set  $V_3$  as the reference node therefore  $V_3 = 0V$ .

(2) Apply KCL and sum the current at nodes  $V_1$  and  $V_2$ , remembering that outgoing current from the nodes is positive

Node 1  $\rightarrow I_1 + I_2 + I_3 + I_4 = 0$   
Node 2  $\rightarrow I_5 + I_6 + I_7 = 0$

(3) Apply all known values and solve for the resulting equations to find the unknowns:

$$\begin{aligned} I_1 &= -5 \text{ A} \\ I_2 &= (V_1 - V_2)/10 \\ I_3 &= 2 \text{ A} \\ I_4 &= V_1/30 \\ I_5 &= (V_2 - V_1)/10 \\ I_6 &= -2 \text{ A} \\ I_7 &= (V_2 - 60)/30 \end{aligned}$$

Apply the known values to earlier KCL equations for Node 1 & 2:

$$\begin{aligned} \text{Node 1} &\rightarrow -5 + (V_1 - V_2)/10 + 2 + V_1/30 = 0 \\ \text{Node 2} &\rightarrow (V_2 - V_1)/10 - 2 + (V_2 - 60)/30 = 0 \end{aligned}$$

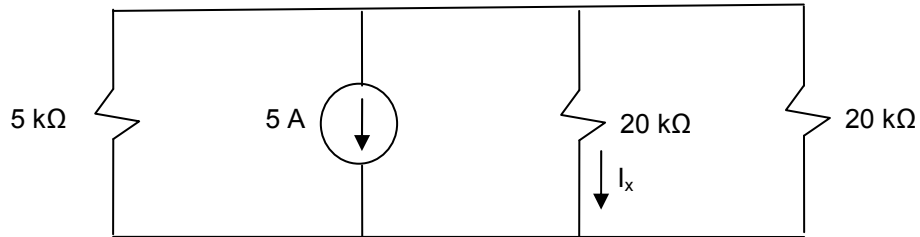
Simplify the above two simultaneous equations:

$$\begin{aligned} \text{Node 1} &\rightarrow 4V_1 - 3V_2 = 90 \\ \text{Node 2} &\rightarrow -3V_1 + 4V_2 = 120 \end{aligned}$$

Solve the above two simultaneous equations to find  $V_1$ . Note that most scientific calculators have matrix functionality which is able to solve the multiple simultaneous equations.

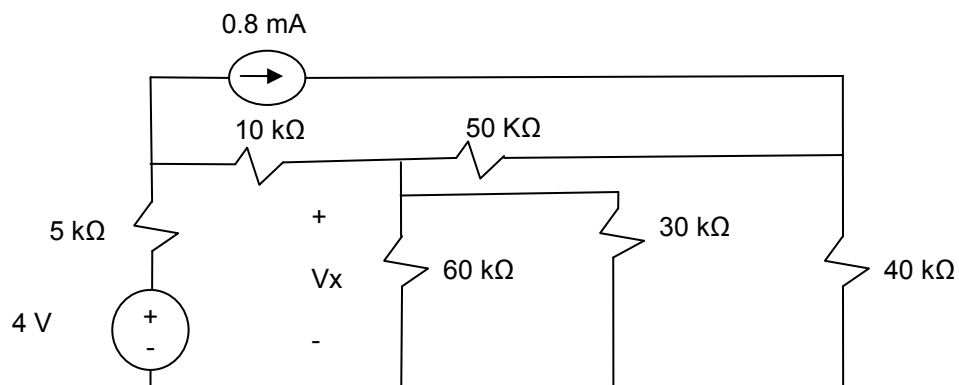
$$V_1 = 102.9 \text{ V} \rightarrow I_p = I_4 = V_1/30 = 3.4 \text{ A}$$

- **Student Exercise** - Kirchoff's Current Law (KCL) Application  
Use KCL to find the value of  $i_x$  in the following circuit:



Solution:

- **Student Exercise** - Use KCL to find the value of  $V_x$  in the following circuit:



Solution

❖ Kirchhoff's Voltage Law (KVL)

Kirchhoff's Voltage Law (KVL) is equivalent of KCL. KVL states that "The algebraic sum of all the voltages around any loop in a circuit equals zero" where:

- A closed path or loop is the trace of a closed path in a circuit through selected basic circuit elements, which returns to the original node without passing through any intermediate node more than once. We are only interested in loops that do not include other loops which are also called Meshes.
- Pick either clockwise or counter-clockwise as positive, but be consistent throughout the circuit analysis!
- Equations generated by Kirchhoff's Voltage Law will be independent only if the selected loop does not include other complete loops.

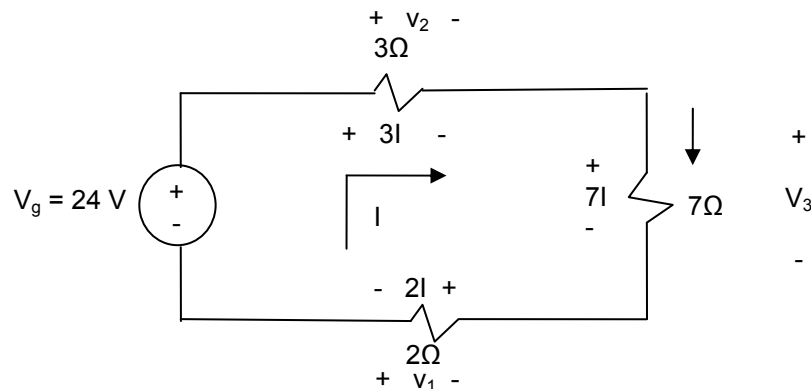
The following three steps are a simplified approach to performing KVL analysis:

- 1) Identify the meshes
- 2) Assign direction to the current in each of the identified meshes (be consistent)
- 3) Write KVL equation for each mesh (sum of all the voltages around a given mesh is equal to zero). Remember Passive Sign Convention (voltage drops is in the direction of current).

The next set of examples are intended to apply the KVL in a step-by-step approach to analyze circuits. It is important to be consistent in applying KVL for improved accuracy and speed of analysis.

➤ Example - Kirchhoff's Voltage Law

Apply Kirchhoff's Voltage Law to find  $I$ ,  $V_1$ ,  $V_2$ ,  $V_3$  and the power delivered to the 24V Source.



Solution:

- (1) Find the Meshes - there is only one Mesh and it does not include another loop.
- (2) Assign a direction to the current (be consistent)
- (3) Assign voltage polarity to elements if there is an element without polarity. Use passive sign convention where voltage drops (+ to -) in the direction of current flow.
- (4) Go around the loop (clockwise) and sum up the voltages which should equal to zero based on Kirchhoff's Voltage Law.

$$-24 + 3I + 7I + 2I = 0 \rightarrow I = 2 \text{ A}$$

Therefore:

$$V_1 = -2I = -4 \text{ (the negative is due to opposite polarity.)}$$

$$V_2 = 3I = 6$$

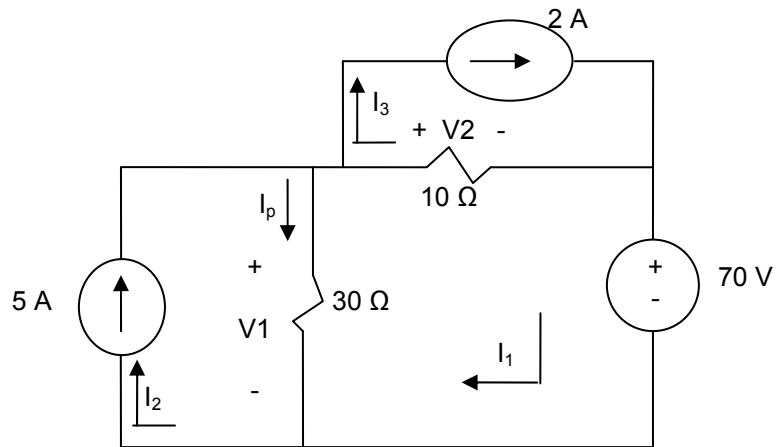
$$V_3 = 7I = 14$$

(5) The only active (power generating) component is the 24V voltage source.

$$\text{Power} = V \cdot I = -(2) \cdot 24V = -48 \text{ W (power delivered)}$$

*Note: It is multiplied by negative since current is entering into the – side of voltage source*

❖ Example – Find  $I_p$  using KVL:



Solution

(1) There are three loops that do not include other loops (3 Meshes)

(2) See diagram

(3) Apply KVL

$$\text{Loop 1} \rightarrow 30(I_1 - I_2) + 10(I_1 - I_3) + 70 = 0 \rightarrow 40I_1 - 30I_2 - 10I_3 = -70$$

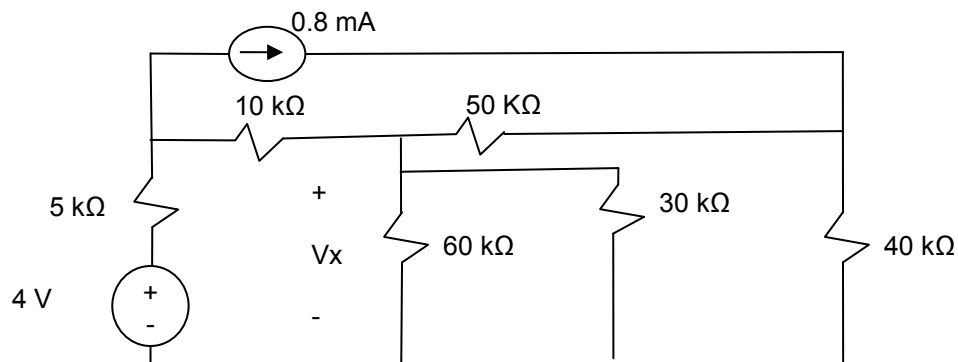
$$\text{Loop 2 "has independent current source"} \rightarrow I_2 = 5A$$

$$\text{Loop 3 "has independent current source"} \rightarrow I_3 = 2A$$

$$\text{Plug known values into Loop 1 equation} \rightarrow 40I_1 - 30 \cdot 5 - 10 \cdot 2 = -70 \rightarrow I_1 = 2.5 \text{ A}$$

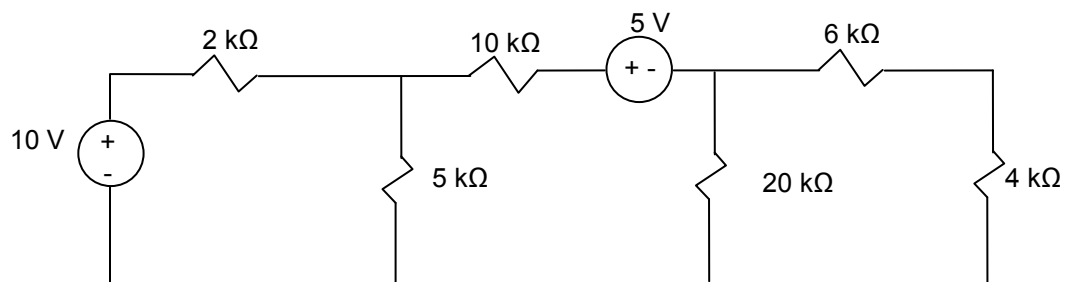
$$\text{Therefore } I_p = I_2 - I_1 = 2.5 \text{ A}$$

➤ **Student Exercise** - Use KVL to find the value of  $V_x$  in the following circuit:



Solution

- ❖ **Student Exercise** – Find the power at 10 v supply using KVL first and check you answer using KCL.

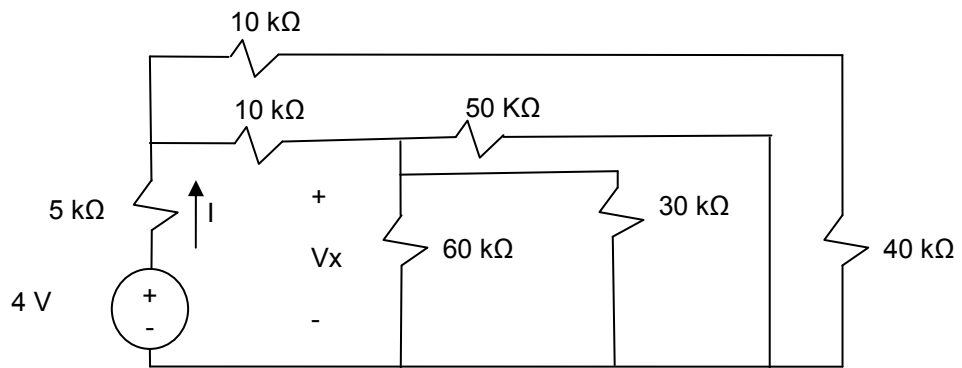


Solution

- ❖ **Student Exercise** – Comparison of KCL, KVL and Simplification analysis techniques

For the following circuit:

- use Resistor simplification to find the value of  $I$ .
- use KCL to find the value of  $I$
- use KVL to find the value of  $I$



Solution

## 4.7. Capacitor, C

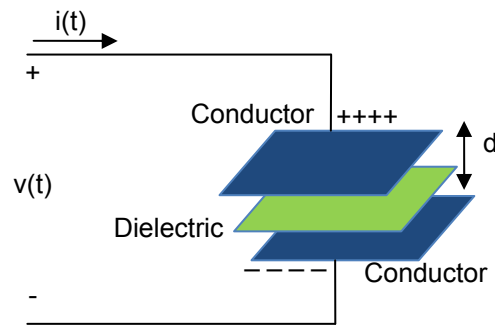
A capacitor is made up of two electrical conductors separated by a dielectric (insulating) material. As voltage varies with time, the displacement of charge also varies with time, causing what is known as the displacement current. The displacement current is proportional with the change in voltage over time.

Below is a basic capacitor construction for the purpose of demonstrating the underlying physical phenomenon and corresponding symbol:

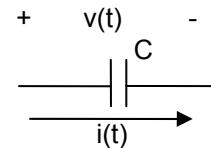
### Physical Construction →

$$C = \frac{\epsilon A}{d} \text{ where}$$

C is capacitance in Farad  
 A is area of conductor plate  
 d is the thickness of dielectric  
 $\epsilon$  the permittivity of the dielectric



### Circuit Symbol →



The following equation relates the capacitor current to voltage change:

$$i = C \frac{dv}{dt}$$

Where:

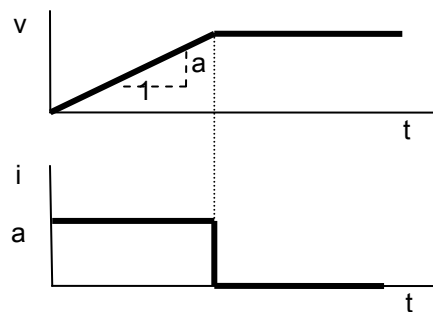
v is in Volts

i is in Amps

C is in Farads

$dv/dt$  is the change in voltage divided by the change in time

The following Chart shows the relationship between the capacitor voltage changes and corresponding current changes:



Observations:

- If voltage is constant, then the current across a capacitor is zero.
- Instantaneous change in voltage ( $dv/dt = \infty$ ) creates a infinite current. Although in real life instantaneous and infinity are not attainable, Capacitors can cause large increases in current due to an abrupt change in voltage.

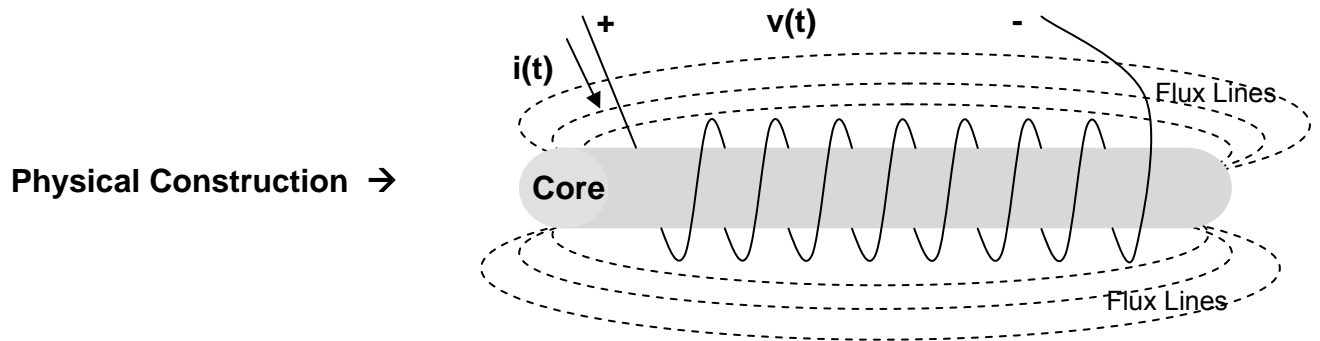
Finally, A capacitor, C, is a passive element; meaning it does not produce power but is able to store energy.



## 4.8. Inductor, L

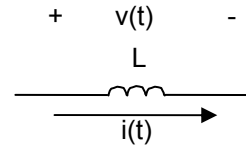
Inductor is traditionally built by winding a coil of wire around a supporting core in which its material may or may not be magnetic. In some cases, the core is simply air (a non-conducting - dielectric). Inductors are manufactured from a variety of materials and are packaged based on the application requirement, so it is difficult to know an inductor's characteristic without the manufacturer's specifications.

Below is a basic inductor construction for the purpose of demonstrating the underlying physical phenomenon and corresponding schematics:



### **Circuit Symbol** →

*Note that passive convention is used.*



The following equation relates the voltage to the change in current:

$$v = L \, di/dt$$

Where:

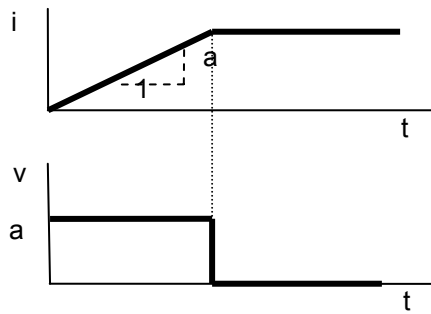
v is in Volts

i is in Amps

L is in Henrys

$di/dt$  is the change in current divided by the change in time

The diagram show below is a graphical representation of the relationship between inductor current and voltage.



Observations:

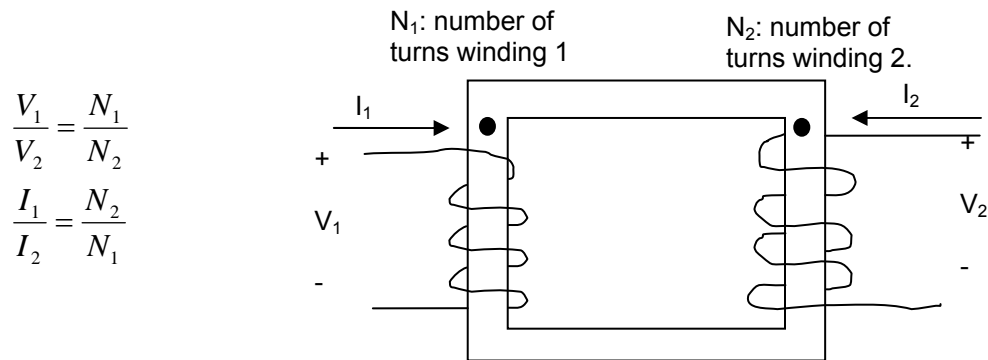
- If the current is constant, then the voltage across an inductor is zero.
- Instantaneous change in current ( $v=di/dt = \infty$ ) creates an infinite voltage. We are not able to create such a change, but we can get close to it. For example, when a switch has been closed for a long time and then it is opened, you may see a spark which is the results of the large voltage created due to a rapid change in current.

An inductor, L, is a passive element; meaning, it does not produce power but it is able to store energy for later release..

## 4.9. Transformers

Transformers are simply two windings near each other, such that the electrical field generated by the change in current in one generates current in the second winding. Commonly a conductive core is used to improve the efficiency of the interaction (coupling) between the two windings.

The relationships between currents and voltages of the two windings in a transformer are shown below:



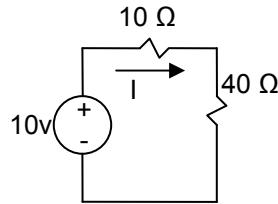
The most common use of transformers is in changing current or voltage by adjusting the ratio of number of turns on the two sets of windings. Transformers are used in a wide variety of electrical systems from refrigerators to computers.

#### 4.10. Additional Resources

- ❖ Nilsson, J. Electrical Circuits. (2004) Pearson.

## 4.11. Problems

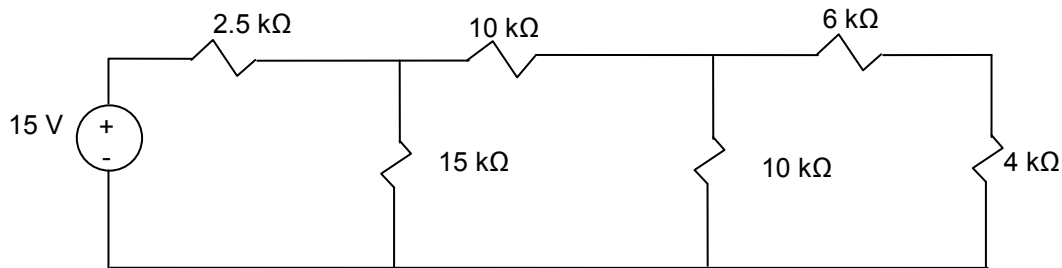
1. For the following circuit:



- Calculate the value of  $I$
- What are the power delivered to each of the resistors and the power generated by the voltage source?
- Is this a valid circuit? Show your reasoning.

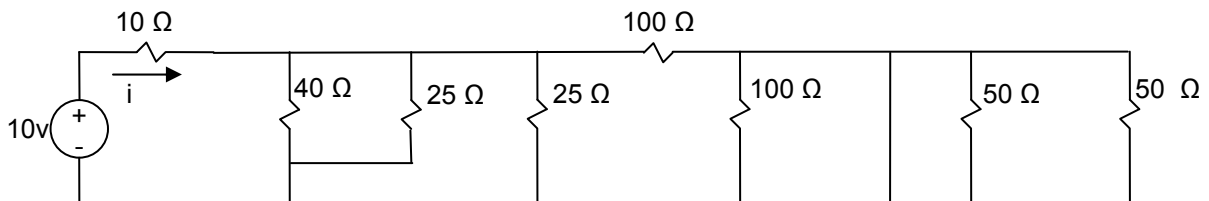
2.

- Simplify the following circuit to an equivalent circuit with one resistor and one voltage.

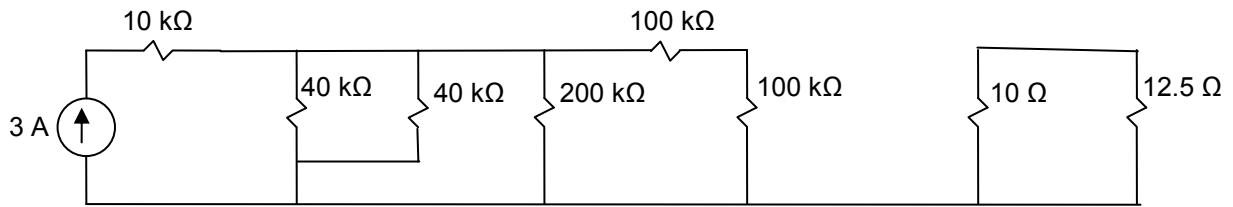


- Use the resulting circuit from part (a) to find the current through the 15 volt source.
- What is the power delivered by the 15 volt source.

3. Use resistor simplification to find the current through 10 Ω resistor.

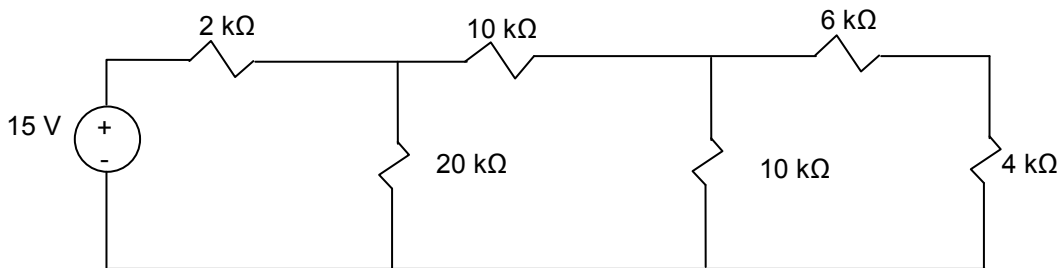


4. Use resistor simplification to find the power through 3 A current source.



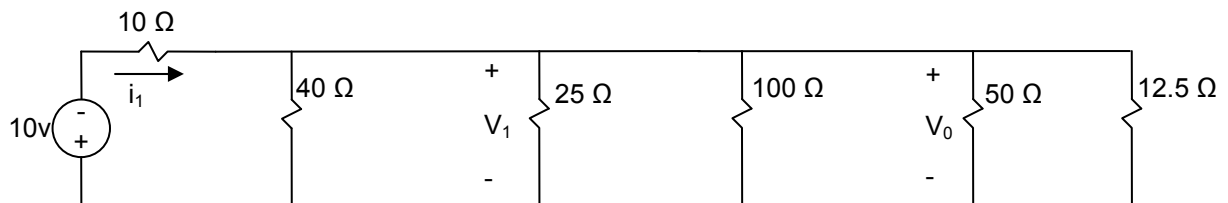
5. For the following Circuit:

- Identify and count the number of essential nodes.
- How many independent equations can you write using KCL approach?
- Use KCL to find the current through 6 kΩ resistor.

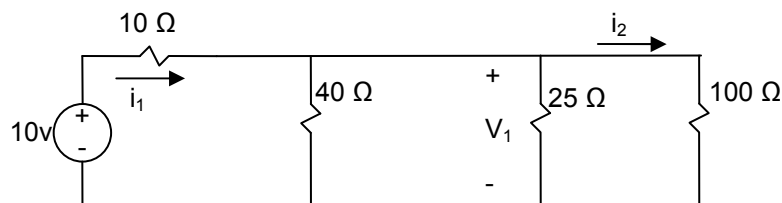


6. For the following Circuit:

- Identify and count the number of essential nodes.
- How many independent equations can you write using KCL approach?
- Use KCL to find the  $I_1$ ,  $V_1$  and  $V_0$ .



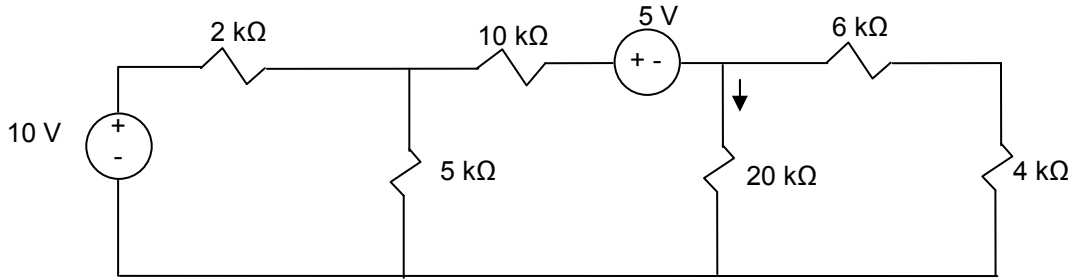
7. Use KCL to calculate the value of  $I_1$ ,  $V_1$  and  $I_2$



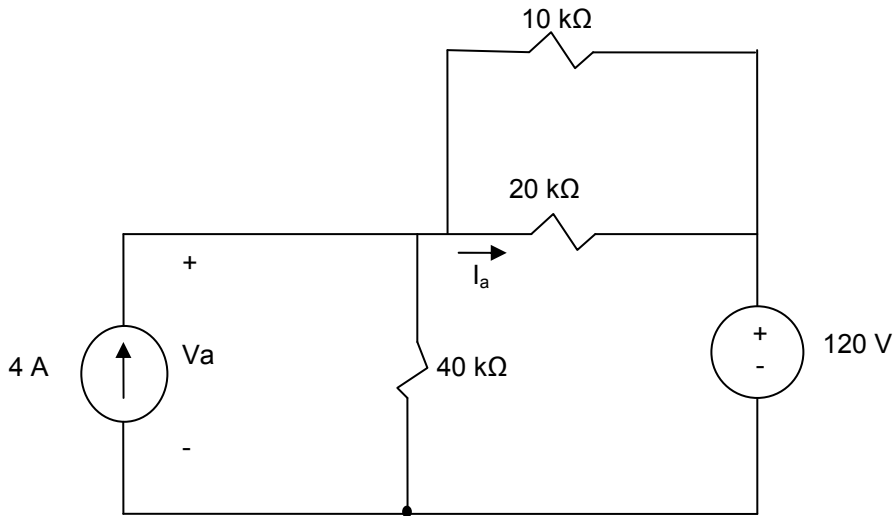
8. For the following Circuit:

- Find the number of loops and meshes.

- b) How many equations can you write using KVL approach  
 c) Find the current through the  $10\text{ k}\Omega$  using KVL.



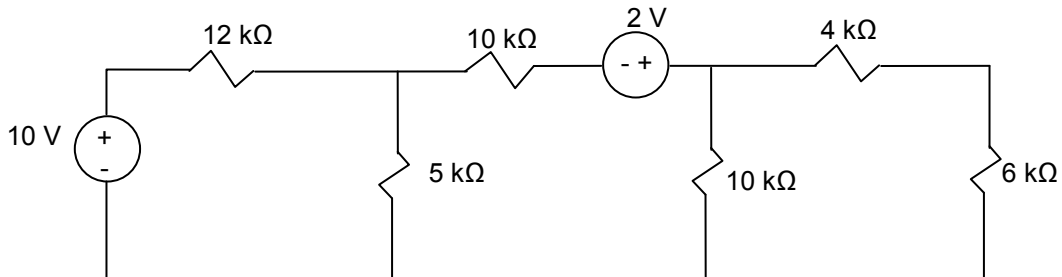
9.



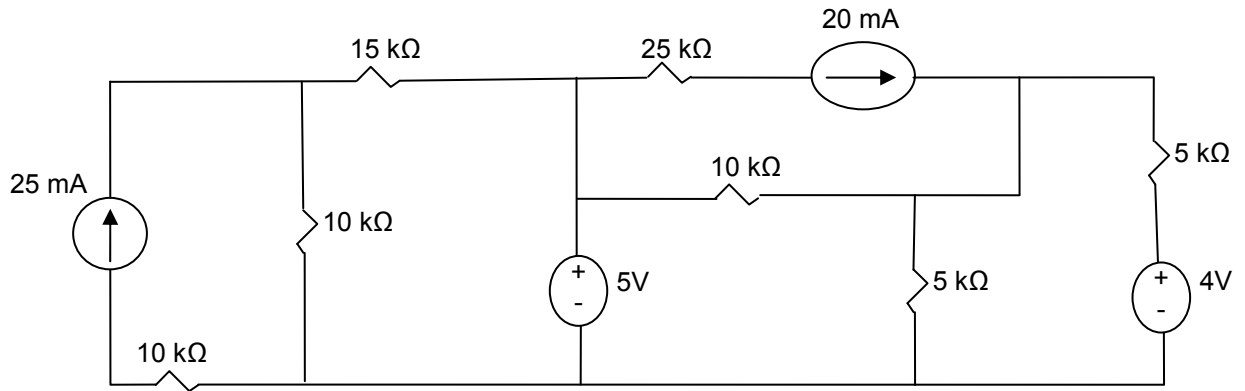
For the above circuit:

- a) Find the number of loops and meshes.  
 b) How many equations can you write using KVL approach  
 c) Find  $I_a$  and  $V_a$  using KVL Approach  
 d) Is this a valid circuit? Show the calculations that support your answer.

10. Use KCL to calculate the voltage across the  $5\text{ k}\Omega$  and the power consumed by  $6\text{ k}\Omega$ .



11. For the following circuit:



- Is this a valid circuit? Explain your answer.
- Identify the node with the highest voltage? Show your work.
- Identify the branch with the highest current? Show your work:



## **Chapter 5. Digital Logic**

### **5.1. Key Concepts and Overview**

- ❖ Digital vs. Analog
- ❖ Digital Design Overview
- ❖ Binary Number Systems
- ❖ Binary Algebra
- ❖ Standard Logic Gates
- ❖ Simple Input and Output Configurations
- ❖ Beyond Standard Logic

## 5.2. Digital vs. Analog

Analog or continuous data have continuous values similar to real numbers. On the other hand, digital or discrete data only have distinct value similar to integer numbers.

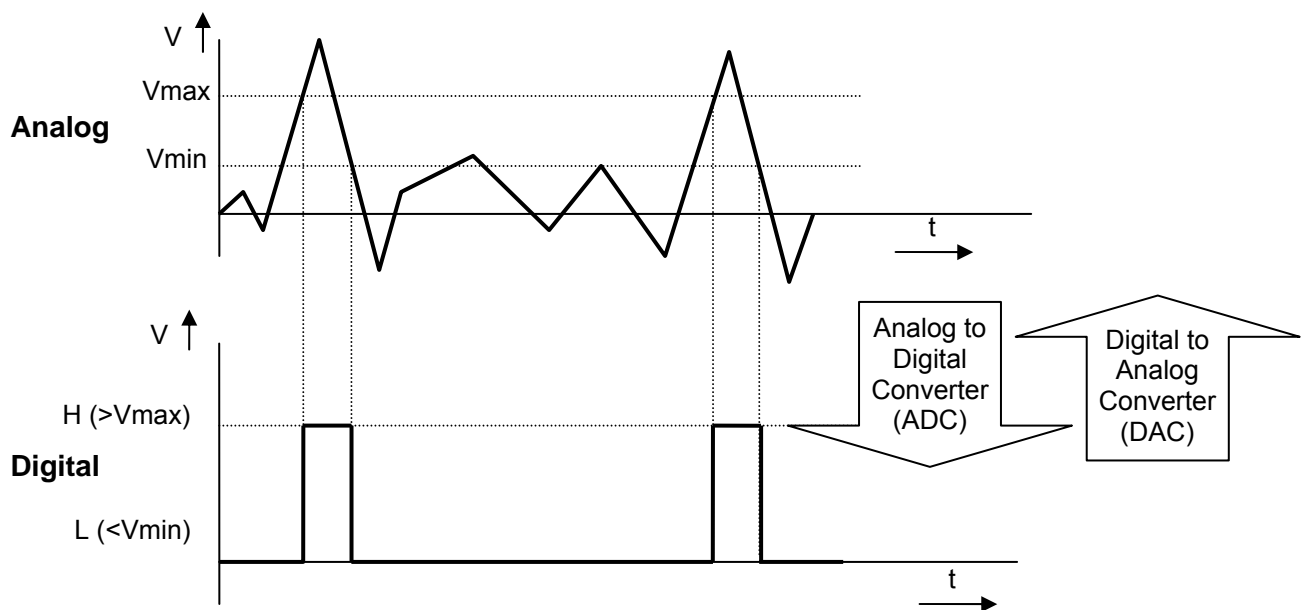
The first step is to convert analog data to digital data. So we have to convert a continuous value to a two value (Binary) system. The following two rules enable the conversion:

- Above a certain voltage level is considered on, high, 1-state or True (H). When ( $V > V_{max}$ ) it is said to be a 1-state or true
- Below a certain voltage level is considered off, low, 0-state or False (L). When ( $V < V_{min}$ ) it is said to be a 0-state or false

The drawback of the digital (Binary) system is that it introduces a discontinuity when a signal goes from 1 to 0 or 0 to 1. This means that the digital value of data is undefined during the transition.

On the other hand digital systems reduce complexity of signals and associated systems. To analyze digital systems, Binary Algebra is all that is needed. Additionally, the cost and complexity of implementing digital systems is significantly lower when compared to a fully analog system.

The following diagram of a human heart beat serves as an example of analog and digital conversion:



❖ Example - Describe the input and output of a traffic intersection in digital form.

Solution:

Car's presence at an intersection:

- \* Car is Present → digital input value "1"
- \* Car is not Present → digital input value is "0"

Status of traffic lights

- \* Red light on → Red = "1"
- \* Red light off → Red = "0"

**Student Exercise** - What are some examples of digital systems? Are the input and output to these systems analog or digital?

### 5.3. Digital Design Overview

This Section introduces the idea of Digital Design from transistors to computer systems. All digital systems from smallest to largest run on a 2-valued system (commonly referred to as binary or digital system), so an electronic solution is needed to represent the two values. This is typically accomplished with a switch that can be on or off. In the early days, mechanical switches were used, followed by vacuum tubes as switches.

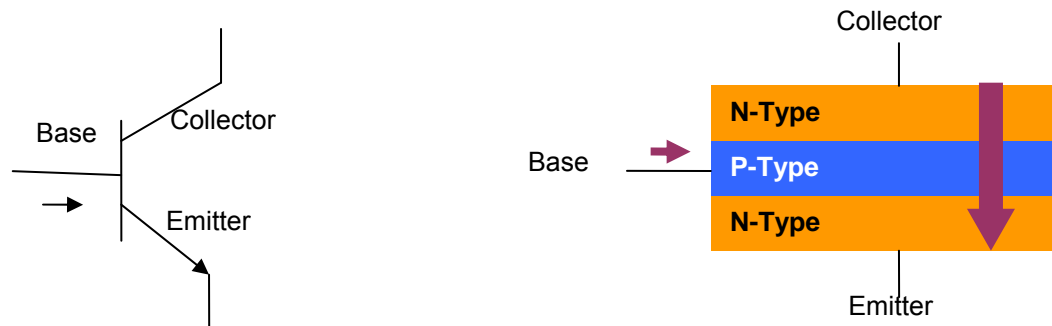
Today, we use transistors that can be configured to approximate the switch on and off modes to build Digital Systems. Transistors are fast (Giga Hertz), low cost (millions for a few dollars) and small (millions per  $\text{cm}^2$ ). Transistors are the reason for the electronic and computer revolution.

Below is some key information about the transistor and its operations:

- The transistor was invented by three scientists at the Bell Laboratories in 1947 and has rapidly replaced the vacuum tube as an electronic signal regulator switch.
- Transistors are the basic elements in integrated circuits (ICs), which consist of very large numbers of transistors interconnected with circuitry and packaged into a single silicon microchip or "chip." A Pentium chip has tens of millions of transistors.
- A transistor is most commonly built on semiconductor material, and it is basically a switch (from a digital point-of-view):

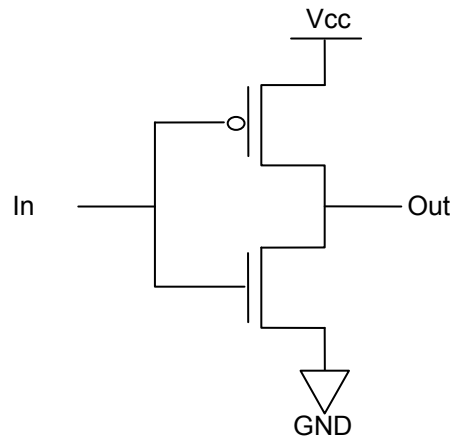
#### NPN Transistor Example

"current at the base causes Collector-Emitter connection change from an open to a short"



The semiconductor material is given special properties by chemical processes. For example doping process adds extra electrons to the material (which is then called N-type for the extra negative charge carriers) or creates "holes" in the material's crystal structure (which is then called P-type because it results in more positive charge carriers).

Today's computers use circuitry implemented with Complementary Metal Oxide Semiconductor (CMOS) technology. CMOS technology always uses two complementary transistors (one with N-type material; the other with P-type material), where one transistor is connected to  $V_{cc}$  and another to Ground:



CMOS' key benefit is low power consumption, since minimal energy is used to maintain a state.

## 5.4. Binary Number Systems

- ❖ Decimal Numbers (base or radix 10)
  - Humans use the Decimal Numbering System as a default, so when you see the number 56, your assumption is that its base or radix is 10 or  $(56)_{10}$  which is “56 base 10”.
  - Each digit is weighted, based on its position in the sequence (power of 10) from the Least Significant Digit (LSD, power of 0) to the Most Significant Digit (MSD, highest power).
  - Each digit must be less than 10 (0 to 9).

For example,  $(2375.46)_{10}$  is evaluated as:

	MSD			LSD		
Digit notation	$d_3$	$d_2$	$d_1$	$d_0$	$d_{-1}$	$d_{-2}$
Digit	2	3	7	5	4	6
Value	$10^3$	$10^2$	$10^1$	$10^0$	$10^{-1}$	$10^{-2}$
Results=Value*Digit	2000	300	70	5	0.4	0.06

$$(2375.46)_{10} = 2000 + 300 + 70 + 5 + 0.4 + 0.06$$

\*\*Note: The general term for a decimal point is *radix point*

- ❖ Binary Number (Base or radix 2)
 

The digital or computer technology is based on the Binary Number System, since the foundation is based on a transistor, which only has two positions: on or off.

Each digit of the number is called a bit or which is a short for **binary digits**. Also, a group of 8-bit is referred to as a byte and group of 4-bit is referred to as a nibble. Here a few key points:

- Each bit is weighted (powers of 2) based on its position in the sequence from the Least Significant Bit (LSB) to the Most Significant Bit (MSB).
- Each bit must be less than 2 (0,1).

For example,  $(1010.11)_2$  is evaluated as:

	MSB			LSB		
Digit notation	$b_3$	$b_2$	$b_1$	$b_0$	$b_{-1}$	$b_{-2}$
Digit	1	0	1	0	1	1
Value	$2^3$	$2^2$	$2^1$	$2^0$	$2^{-1}$	$2^{-2}$
Results=Value*Digit	8	0	2	0	0.5	0.25

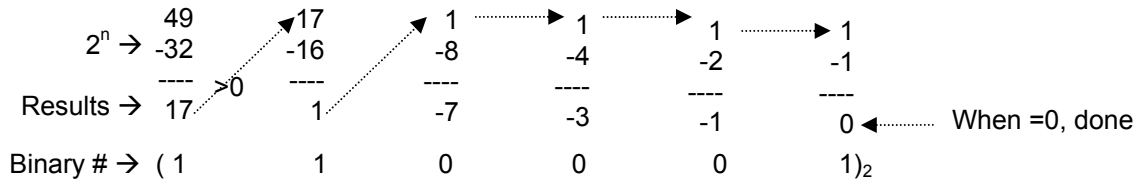
$$(1010.11)_2 = 8 + 0 + 2 + 0 + 0.5 + 0.25 = (10.75)_{10}$$

\*\*Note: The general term for a decimal point is radix point

- In Binary, the count starts at 0 (called 0-referencing) where as in Decimal, the count typically starts with 1 (called 1-referencing)
- ❖ Decimal to Binary Conversion – “Subtract the weight method”
  - Steps:
    - (1) Find the largest power of 2 ( $2^n$ ) that can be subtracted out of the decimal number

- (2) Take the result and subtract ( $2^{n-1}$ ) from it  
 (a) If the result is not negative then that bit is one  
 (b) If the result is negative, then that bit is zero and the result equals the result from step 1  
 (3) Repeat step 2 until the result is exactly 0

➤ Example: convert  $(49)_{10}$  to a binary number



❖ Binary to Decimal Conversion – “Add the weight method”

➤ Step:

- (1) Simply multiply each bit with its weight and add to get the decimal number

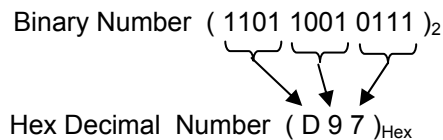
➤ Example: Convert  $(110001)_2$  to a decimal number

$$(110001)_2 = (1 * 2^5 + 1 * 2^4 + 0 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0)_{10} = (49)_{10}$$

❖ Hexadecimal (base 16 or Hex)

The Hexadecimal numbers are used by humans to represent long strings of bits, since they are:

- Easy to convert (groups of 4-bit convert to one Hex digit)



*Note that in the Hexadecimal Number Systems, each digit may have value from 0 to 15. 0 to 9 are represented as 0 to 9 and 10 to 15 are represented by A, B, C, D, E and F.*

- Easy to read, for example  $(0E7)_{Hex}$  is easier to read than  $(011100111)_2$ .

❖ Binary Codes

Information has to be translated into Binary in order to be processed by Digital Systems. Some of the most commonly used binary codes are the ones that translation language characters into binary code. The minimum number of bits required to represent all the keys on a keyboard can be calculated based on the following inequality:

$$2^{\text{Number of Bits}} \geq \text{Number of keys}$$

There are a number of Binary codes that are used to represent numbers and the alphabet such as:

- Binary Coded Decimal (BCD) – used to represent each decimal digit with a fixed number of bits
- Extended Binary-Coded Decimal Interchange Code (EBCDIC) – IBM’s way of representing alphanumeric and control characters.
- American Standard Code for Information Interchange (ASCII) Code is used to represent alphanumeric and control characters in 8-bit. Unicode is a 16-bit derivative of ASCII code

that is implemented in most computers. With 16-bit, Unicode allows characters to set a larger than 256 characters (such as the Chinese Character Set) to be represented.

By far, Unicode (a derivative of ASCII code) is the most commonly used coding system for language character translation. The following table shows the equivalent binary number for each character.

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	<b>NUL</b> (null)	32	20	040	&#32;	Space	64	40	100	&#64;	@	96	60	140	&#96;	`
1	1	001	<b>SOH</b> (start of heading)	33	21	041	&#33;	!	65	41	101	&#65;	A	97	61	141	&#97;	a
2	2	002	<b>STX</b> (start of text)	34	22	042	&#34;	"	66	42	102	&#66;	B	98	62	142	&#98;	b
3	3	003	<b>ETX</b> (end of text)	35	23	043	&#35;	#	67	43	103	&#67;	C	99	63	143	&#99;	c
4	4	004	<b>EOT</b> (end of transmission)	36	24	044	&#36;	\$	68	44	104	&#68;	D	100	64	144	&#100;	d
5	5	005	<b>ENQ</b> (enquiry)	37	25	045	&#37;	%	69	45	105	&#69;	E	101	65	145	&#101;	e
6	6	006	<b>ACK</b> (acknowledge)	38	26	046	&#38;	&	70	46	106	&#70;	F	102	66	146	&#102;	f
7	7	007	<b>BEL</b> (bell)	39	27	047	&#39;	'	71	47	107	&#71;	G	103	67	147	&#103;	g
8	8	010	<b>BS</b> (backspace)	40	28	050	&#40;	(	72	48	110	&#72;	H	104	68	150	&#104;	h
9	9	011	<b>TAB</b> (horizontal tab)	41	29	051	&#41;	)	73	49	111	&#73;	I	105	69	151	&#105;	i
10	A	012	<b>LF</b> (NL line feed, new line)	42	2A	052	&#42;	*	74	4A	112	&#74;	J	106	70	152	&#106;	j
11	B	013	<b>VT</b> (vertical tab)	43	2B	053	&#43;	+	75	4B	113	&#75;	K	107	71	153	&#107;	k
12	C	014	<b>FF</b> (NP form feed, new page)	44	2C	054	&#44;	,	76	4C	114	&#76;	L	108	72	154	&#108;	l
13	D	015	<b>CR</b> (carriage return)	45	2D	055	&#45;	-	77	4D	115	&#77;	M	109	73	155	&#109;	m
14	E	016	<b>SO</b> (shift out)	46	2E	056	&#46;	.	78	4E	116	&#78;	N	110	74	156	&#110;	n
15	F	017	<b>SI</b> (shift in)	47	2F	057	&#47;	/	79	4F	117	&#79;	O	111	75	157	&#111;	o
16	10	020	<b>DLE</b> (data link escape)	48	30	060	&#48;	0	80	50	120	&#80;	P	112	76	160	&#112;	p
17	11	021	<b>DC1</b> (device control 1)	49	31	061	&#49;	1	81	51	121	&#81;	Q	113	77	161	&#113;	q
18	12	022	<b>DC2</b> (device control 2)	50	32	062	&#50;	2	82	52	122	&#82;	R	114	78	162	&#114;	r
19	13	023	<b>DC3</b> (device control 3)	51	33	063	&#51;	3	83	53	123	&#83;	S	115	79	163	&#115;	s
20	14	024	<b>DC4</b> (device control 4)	52	34	064	&#52;	4	84	54	124	&#84;	T	116	80	164	&#116;	t
21	15	025	<b>NAK</b> (negative acknowledge)	53	35	065	&#53;	5	85	55	125	&#85;	U	117	81	165	&#117;	u
22	16	026	<b>SYN</b> (synchronous idle)	54	36	066	&#54;	6	86	56	126	&#86;	V	118	82	166	&#118;	v
23	17	027	<b>ETB</b> (end of trans. block)	55	37	067	&#55;	7	87	57	127	&#87;	W	119	83	167	&#119;	w
24	18	030	<b>CAN</b> (cancel)	56	38	070	&#56;	8	88	58	130	&#88;	X	120	84	170	&#120;	x
25	19	031	<b>EM</b> (end of medium)	57	39	071	&#57;	9	89	59	131	&#89;	Y	121	85	171	&#121;	y
26	1A	032	<b>SUB</b> (substitute)	58	3A	072	&#58;	:	90	5A	132	&#90;	Z	122	86	172	&#122;	z
27	1B	033	<b>ESC</b> (escape)	59	3B	073	&#59;	;	91	5B	133	&#91;	[	123	87	173	&#123;	{
28	1C	034	<b>FS</b> (file separator)	60	3C	074	&#60;	<	92	5C	134	&#92;	\	124	88	174	&#124;	
29	1D	035	<b>GS</b> (group separator)	61	3D	075	&#61;	=	93	5D	135	&#93;	]	125	89	175	&#125;	}
30	1E	036	<b>RS</b> (record separator)	62	3E	076	&#62;	>	94	5E	136	&#94;	^	126	90	176	&#126;	~
31	1F	037	<b>US</b> (unit separator)	63	3F	077	&#63;	?	95	5F	137	&#95;	_	127	91	177	&#127;	DEL

Source: [www.asciitable.com](http://www.asciitable.com)



## 5.5. Binary Algebra

### ❖ Purpose of Theorems

The Theorems and Huntington's Postulates are key in our ability to reduce the number of literal (occurrences of variables) used in a function and therefore reduce the number of gates required to implement a given function. Sometimes they are used to simply rearrange the expression so it is easier to implement.

- Example:  $(X+Y) \cdot (X+\bar{Y}) = X$   
It is clear that the right-hand side requires fewer gates to implement than the left-hand side

### ❖ Two methods available for proving theorems

- Prove through Boolean Algebra  
Use Huntington's Postulates or Theorems already proved to show that both sides of theorem are the same.
- Prove through Truth Tables  
Show that for all possible values on the left hand-side are equal to the right-hand side of the equation. This method works well for a small number of variables.

### ❖ Theorems and Proofs

- Theorem 1 "Double Complementation or Double Negation Theorem"

a)  $\overline{\overline{X}} = X$

- Theorem 2 "Idempotency Theorem"

a)  $X+X = X$

b)  $X \cdot X = X$

- Theorem 3 "Identity Element Theorem"

a)  $X + 1 = 1$

b)  $X \cdot 0 = 0$

- Theorem 4 "Absorption Theorem"

a)  $X + X \cdot Y = X$

b)  $X \cdot (X + Y) = X$

- Theorem 5 "Associative Theorem"

a)  $X + (Y + Z) = (X + Y) + Z$

b)  $X \cdot (Y \cdot Z) = (X \cdot Y) \cdot Z$

- Theorem 6 "Adjacency Theorem"

a)  $X \cdot Y + X \cdot \bar{Y} = X$

b)  $(X + Y) \cdot (X + \bar{Y}) = X$

*"This theorem is especially useful in minimizing expressions"*

- Theorem 7 "Consensus Theorem"

a)  $X \cdot Y + \bar{X} \cdot Z + Y \cdot Z = X \cdot Y + \bar{X} \cdot Z$

b)  $(X + Y) \cdot (\bar{X} + Z) \cdot (Y + Z) = (X + Y) \cdot (\bar{X} + Z)$

- Theorem 8 "Simplification Theorem"

a)  $X + \bar{X} \cdot Y = X + Y$

$$b) X \cdot (\overline{X} + Y) = X \cdot Y$$

➤ Theorem 9 “DeMorgan’s Theorem (2-Variable form)”

$$a) \overline{X + Y} = \overline{X} \cdot \overline{Y}$$

$$b) \overline{X \cdot Y} = \overline{X} + \overline{Y}$$

➤ Theorem 10 “DeMorgan’s Theorem (General form)”

$$a) \overline{X_1 + X_2 + \dots + X_n} = \overline{X_1} \cdot \overline{X_2} \cdot \dots \cdot \overline{X_n}$$

$$b) \overline{X_1 \cdot X_2 \cdot \dots \cdot X_n} = \overline{X_1} + \overline{X_2} + \dots + \overline{X_n}$$

❖ Example of two types of proofs (Truth Table and Algebraic)

➤ Prove Theorem 8 “Simplification Theorem”

Hint: use a truth table

➤ Prove Theorem 10 “DeMorgan’s Theorem (General form)”

Hint: Apply theorem 9.

## 5.6. Standard Logic Gates

❖ The rest of this Section relies on two-valued Boolean Algebra, i.e.  $B = \{0, 1\}$

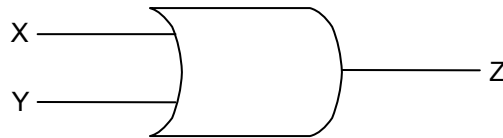
➤ We use variables  $X, Y, Z, A, B, \dots$  and constants 0 and 1

*Note: "0" and "1" are also called identity elements.*

➤ Binary operators

- "+" called "OR"

(1) "OR" symbol  $\rightarrow Z = X + Y$



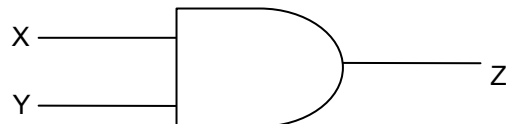
(2) "OR" truth table

X	Y	X+Y
0	0	0
0	1	1
1	0	1
1	1	1

(3) Review the 74LS32 Data Sheet on the website

- "." called "AND"

(1) AND symbol  $\rightarrow Z = X.Y$



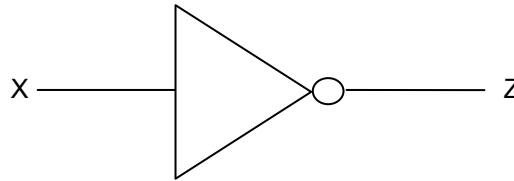
(2) "AND" truth table

X	Y	X.Y
0	0	0
0	1	0
1	0	0
1	1	1

(3) Review 74LS08 Data sheet on the website

- $\bar{\quad}$ ,  $\sim$  Called "NOT, inversion, negation or complement"

(1) "NOT" symbol  $\rightarrow Z = \overline{X} = \sim A = X'$



(2) "NOT" Truth Table

X	Z=X'
0	1
1	0

(3) Review 74LS04 Data Sheet on the website

➤ Order of Precedence

Highest to lowest order of Precedence for Binary Operator: "=", "()", "¯", "·", "+"

Note:

- Parentheses are used to force the operation order sequence, much like in Decimal Algebra.
- The equal sign "=" is an assignment, the same as decimal algebra

➤ An expression is a combination of variables and binary operators  $Z + X \cdot Y + X$

➤ A literal is an occurrence of a variable in an expression

For example  $f(x,y,z) = x + y \cdot x \cdot z + x' \cdot y' \cdot z$  is said to have 7 literals. The number of literals is typically used as a measure of implementation complexity.

❖ Additional standard logic gates

➤ NOR is a negated OR gate

Review 74LS02 Data sheet on the website

➤ NAND is a negated AND gate

Review 74LS00 Data sheet on the website

➤ XOR "also called Module 2 add or exclusive or"  $\rightarrow X \oplus Y = \overline{X} \cdot Y + X \cdot \overline{Y}$

- "XOR" Truth Table

X	Y	$X \oplus Y$
0	0	0
0	1	1
1	0	1
1	1	0

- Review 74LS86 Data sheet on the website

- When using three variables, the operation is performed two at a time as shown below:

$$X \oplus Y \oplus Z = (X \oplus Y) \oplus Z$$

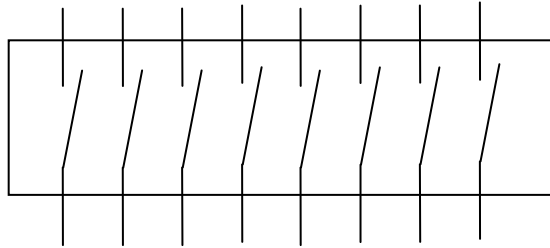
- "XOR" is commonly used to check if there are odd or even numbers of "1". This check is called parity. Odd parity is when there are odd numbers of "1" and even parity is when there are even numbers of "1". Parity check is used for single bit error detection.

## 5.7. Simple Input and Output Configurations

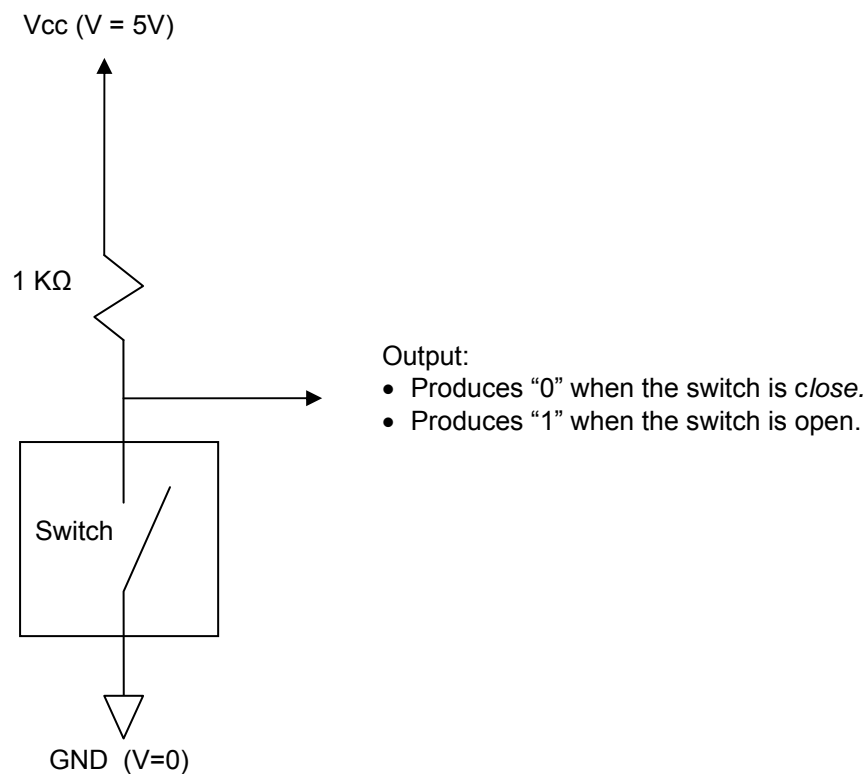
In order use these devices, at least one input of one output is needed. Light Emitting Diodes (LEDs) and Switches are good tools to simulate input and output. This Section outlines the use of switch to produce a 0 or 1 as an input to the digital device and use of LED to display 1 or 0 out as light or no light.

### ❖ Switch as an Input

Switch is either closed (conducting electricity) or open (not conducting). In the Digital System, it is common to use multi-switch packs. Below is the configuration of 8-Switch Dual In-line package:

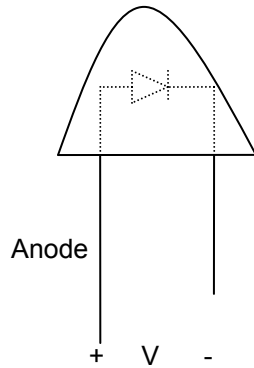


The follow switch configuration produces a digital “1” when the switch is open and produces a digital “0” when the switch is closed.

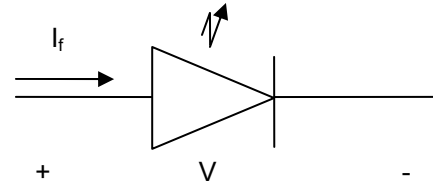


❖ Light Emitting Diode (LED) as an Output

Light Emitting Diode (LED) is used as indicator in many applications from power on/off light to traffic signal lights. LED lamination, current and power specifications vary, depending on design and application. One common LED is the SSL-LX5093LXX, manufactured by Lumex (refer to course website for more details):



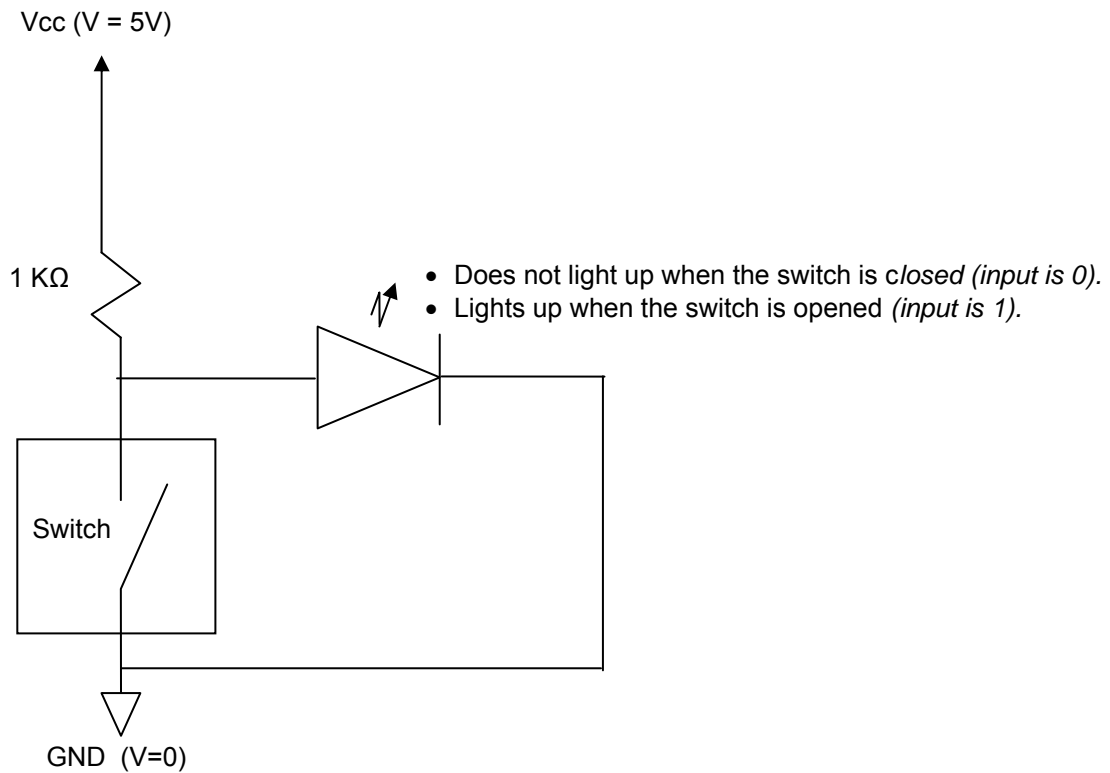
**Packaging Configuration**



**Functional Diagram**

**Rating:**  $I_f < 30 \text{ mA}$  at 2.5 Volts  
**Typical:** +5 V at  $I = 5 \text{ ma}$

Below is an example of using LED to Display the value of input which is produced by a switch:

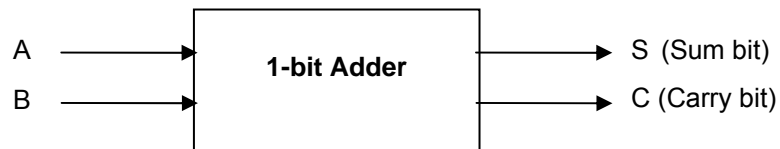


## 5.8. Logic Design Examples

- ❖ Example – Design a 1-bit adder circuit that accepts two digital inputs (A and B) and display the output sum (S) and carry out (C):

Solution

### Step 1. System Diagram (Input & Output Definition)



### Step 2. Truth Table (Input and Output Relationships)

Input		Output	
A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

### Step 3. Output Function (if possible apply Adjacency Theorem to minimize)

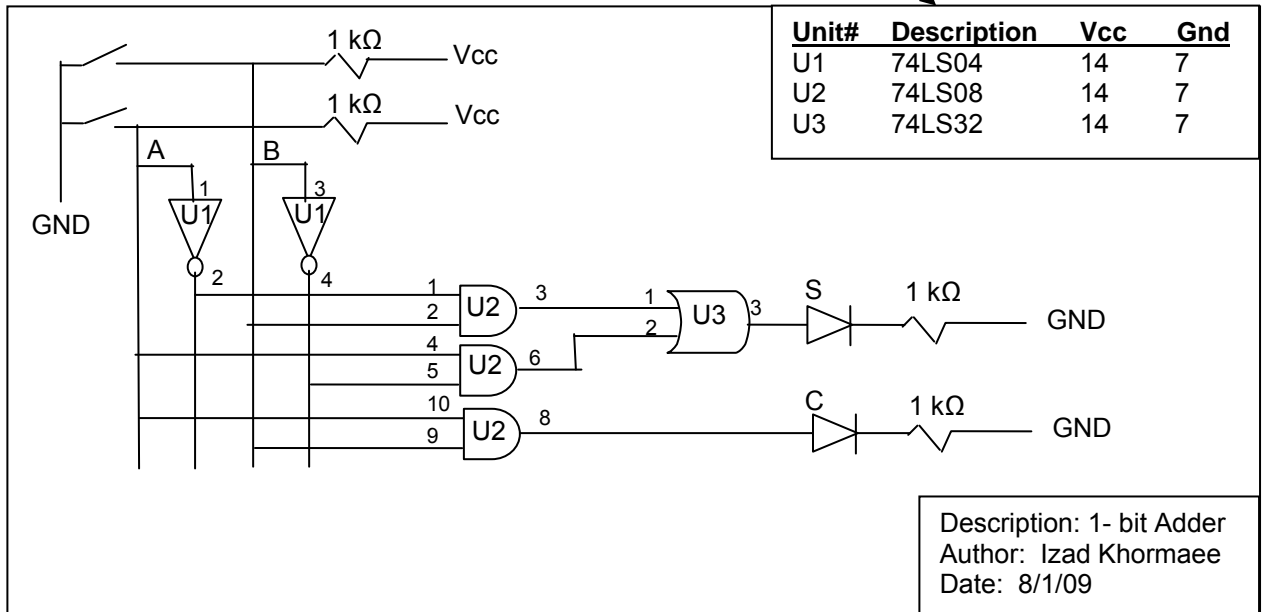
$$S = A'.B + A.B'$$

$$C = A.B$$

**Step 4. Schematics (Component Level Diagram)**

Note: schematics must be on separate sheet of paper and include project and component identification blocks. Typically, landscape is preferred layout for schematics.

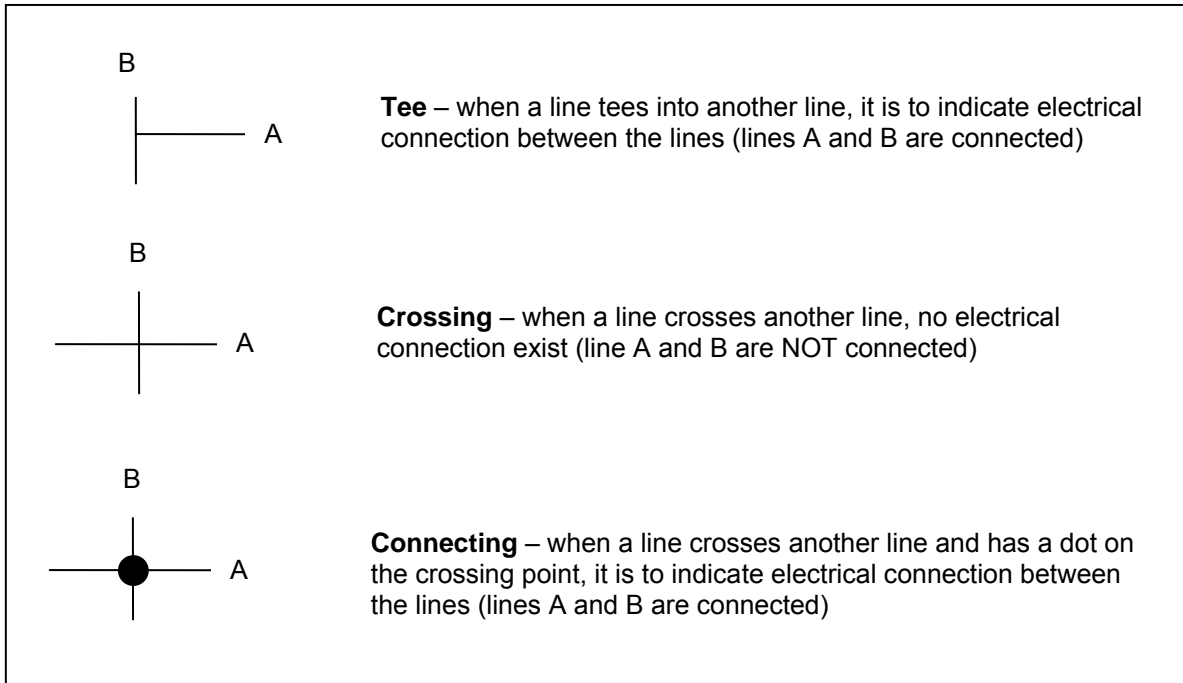
“Component ID block”



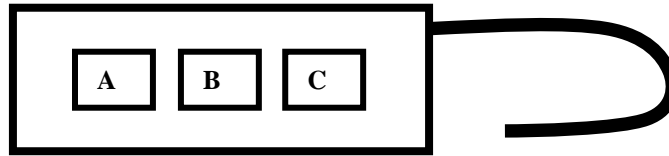
“Project ID block”



Note - the following diagrams show three possible wire crossing and connections between lines in a schematic :

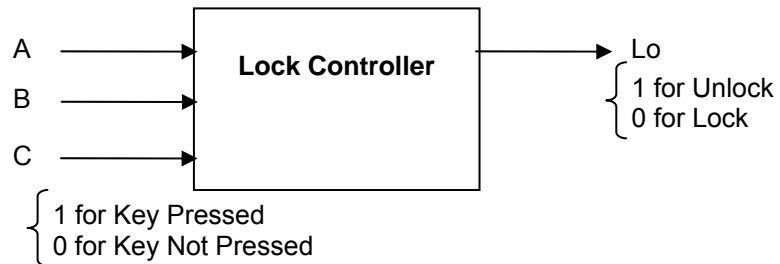


- ❖ Example – Design a 3 button lock that opens when at least two adjacent buttons are pressed.



### Solution

#### Step 1. System Diagram (Input & Output Definition)



#### Step 2. Truth Table (Input and Output Relationships)

Input			Output
A	B	C	Lo
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

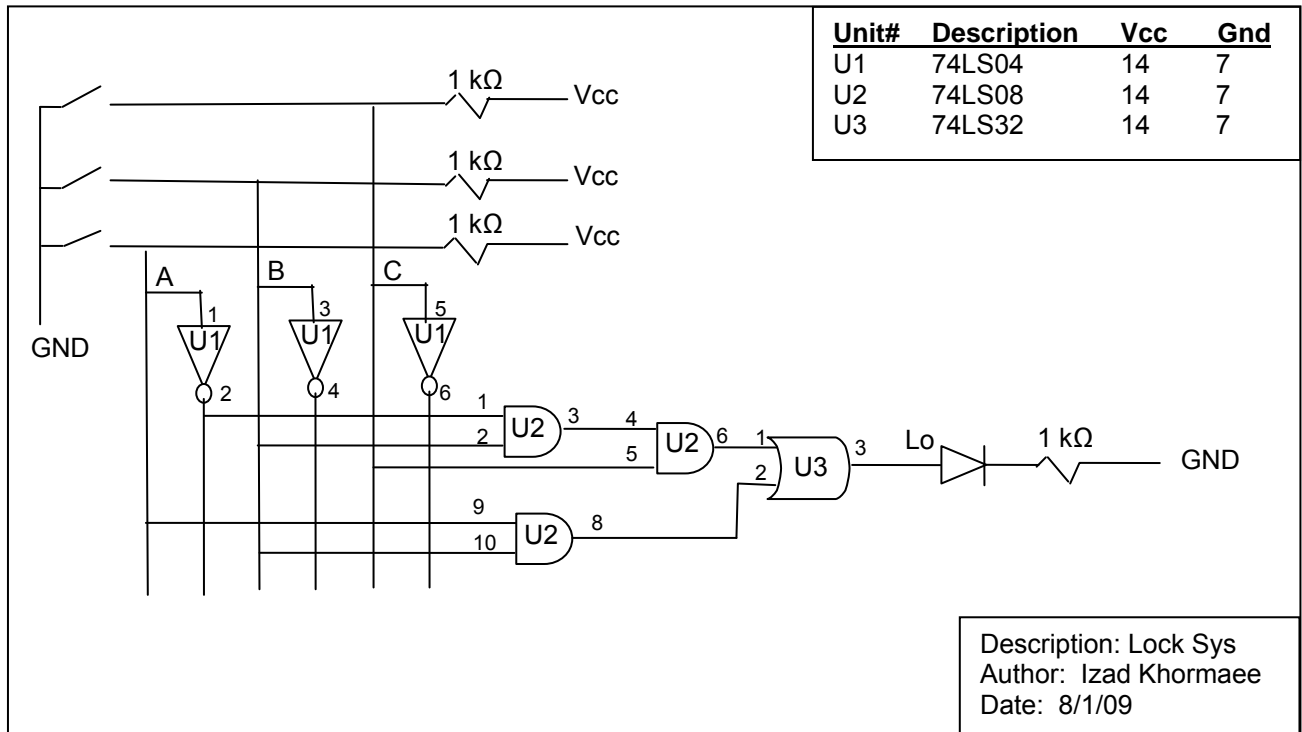
#### Step 3. Output Function (Apply Adjacency Theorem to Minimize)

$$Lo = A'BC + ABC' + ABC$$

Using Adjacency Theorem, we can simplify the above equation to:

$$Lo = BC + AB$$

#### Step 4. Schematics (Component Level Diagram)



- ❖ Student Exercise – Design a circuit that accepts two digital inputs ( $A_1$  and  $A_0$ ) and light up two LED's ( $L_1$  and  $L_0$ ) as described by the following table:

$A_1$	$A_0$	$L_1$	$L_0$
0	0	on	off
0	1	on	on
1	0	off	off
1	1	off	on

Solution

**Step 1. System Diagram (Input & Output Definition)**

**Step 2. Truth Table (Input and Output Relationships)**

**Step 3. Output Function (Apply Adjacency Theorem to Minimize)**

**Step 4. Schematics (Component Level Diagram)**

- ❖ Student Exercise – Design a circuit that accepts three inputs A, B and C and if A=1 then turns on LED La regardless of other inputs. LED Lb will turn on only when A=0 and B=1 regardless of value of C. Finally LED Lc turns on only if A= B=0 and C=1.

**Solution:** In-Class Exercise

**Step 1. System Diagram (Input & Output Definition)**

**Step 2. Truth Table (Input and Output Relationships)**

**Step 3. Output Function (Apply Adjacency Theorem to Minimize)**

**Step 4. Schematics (Component Level Diagram)**

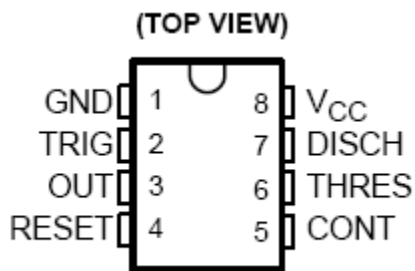
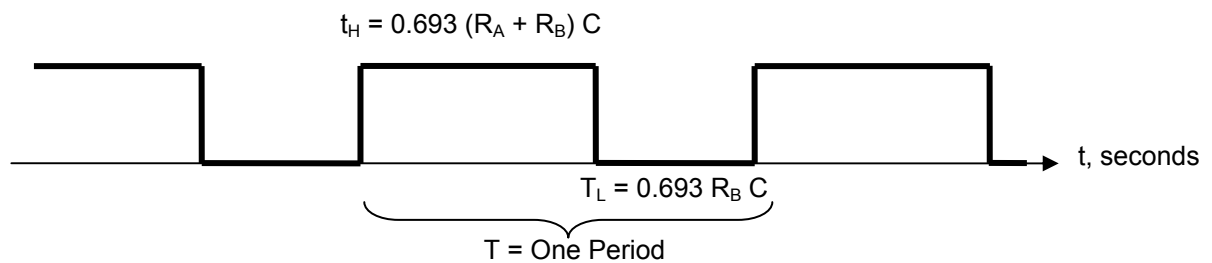
## 5.9. Beyond Standard Logic

There are numerous components that are designed for specific functions and are commercially available. This Section provides descriptions of few as examples, but there are many more available on the course website, and even more in the market.

❖ **Clock Generator Circuit (NE 555 timer)**

Function: NE 555 timer is commonly used to generate a square wave output that is used as a clock input into digital circuits.

Below is the NE 555 configuration that outputs a square wave. This type of output can be used as a clock into other ICs. The pulse width and period can be modified by changes in values of C,  $R_A$  and  $R_B$ .

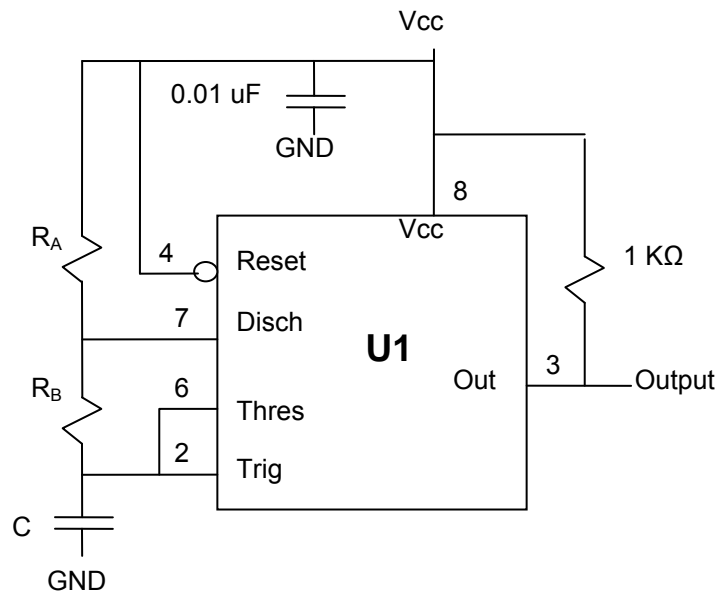


Device #	Pin#	Value
U1	1	GND
U1	3	Out
U1	8	Vcc
U1	5	Open

$$t_H = 0.693 (R_A + R_B) C$$

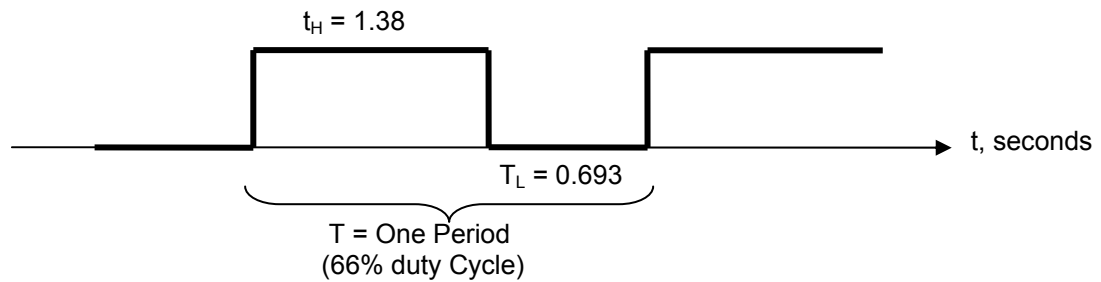
$$t_L = 0.693 (R_B) C$$

The following schematic shows the NE 555 circuit to produce a square wave (typically referred to as clock):



- Example – Use 555 Timer chip to design a circuit that produces a square wave (clock) with approximately  $t_H = 1.38$  seconds and  $T_L = 0.69$  seconds.

**Solution:**



Use the equation to calculate the value of the components:

$$t_H = 0.693 (R_A + R_B) C$$

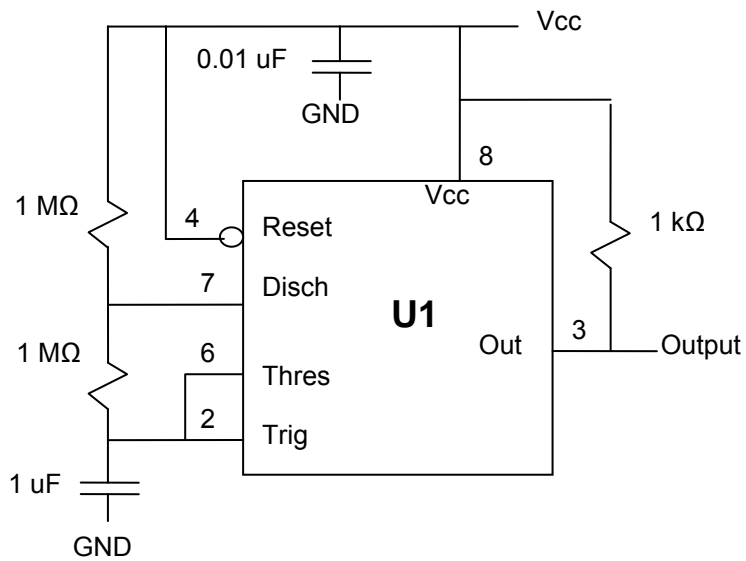
$$t_L = 0.693 (R_B) C$$

Given value of  $t_H = 1.38$  seconds and  $T_L = 0.69$  seconds, we can find:

$$R_A = 1 \text{ M}\Omega$$

$$R_B = 1 \text{ M}\Omega$$

by picking C to be equal to “1 uF”.



- ❖ Student Exercise  
Design a circuit using 555 Timer that generates a clock (square wave) with period of 19 mSec and 60% duty cycle.

**Solution:** In-Class Exercise



## 5.10. Additional Resources

- ❖ Wakerly, I. Digital Design. (2006) Prentice Hall
- ❖ Katz, R. Contemporary Logic Design. (2005) Pearson.
- ❖ Sandige, R. Digital Design Essentials. (2002) Prentice Hall.

## 5.11. Problems

---

1. Convert the following binary numbers to decimal equivalent value. Check your work by converting the number back to its original base after every conversion:

- a)  $(1101)_2$
- b)  $(101001)_2$
- c)  $(11001010)_2$
- d)  $(10001100)_2$

---

2. Convert the following decimal numbers to binary equivalent value. Check your work by converting the number back to its original base after every conversion:

- a)  $(35)_{10}$
- b)  $(139)_{10}$
- c)  $(425)_{10}$

---

3. Convert the following numbers to equivalent binary and decimal values:

- a)  $(291)_H$
- b)  $(CAB)_H$

---

4 Write the Hexadecimal equivalent of the text string "Hello World" in ASCII code.

---

5. Design a lock with 4 keys that will open when exactly two of the four keys are pressed. Clearly show the system diagram, truth table, output functions and schematics for your design.

---

6. Design a 2-bit adder that accepts the operand from switches and displays the results using LEDs. Clearly show the system diagram, truth table, output functions and schematics for your design.

---

7. Design a circuit that generates a clock signal with 200 msec period and 75% duty cycle (150 msec high and 50 msec. low).

## **Chapter 6. Computer Architecture and Programming Fundamentals**

### **6.1. Key Concepts and Overview**

- ❖ Computer Architecture
- ❖ Programming Levels
- ❖ Software Development Steps
- ❖ Development Environment Interface and Structure
- ❖ Using the MATLAB Command Window
- ❖ Creating and Editing MATLAB M-files
- ❖ MATLAB Arithmetic and Logic Operators
- ❖ MATLAB Data Flow Controls

## 6.2. Computer Architecture

The word “architecture” is used here to describe the relationships of major components of a computer much like a home plan describes the architecture of a home. In this section, architecture will be discussed in terms of Design Considerations, Software Layers and Hardware.

### ❖ Computer Design Considerations

Over time important of various design considerations have changed. But three main areas have remained “Speed/Performance”, “Maintainability and Reliability” and “Hardware Cost and Memory Requirements”.

#### ➤ Speed/Performance

Today, performance continues to be an important issue as applications has increased in complexity due to Increased demands for graphics, database, and operating systems. Designer have attempted to answer the needs by:

- Higher Performance processors and more memory
- Memory hierarchy (executable memory, cache, secondary storage)
- Parallel computing
- Improve a software and hardware design processes

#### ➤ Maintainability and Reliability

As systems become more complicated, the need for maintainability and expandability becomes increasingly more important.

#### ➤ Hardware Cost and Memory Requirement

As technology advances, the issue of hardware cost and memory use minimization is less of an issue in designing computers.

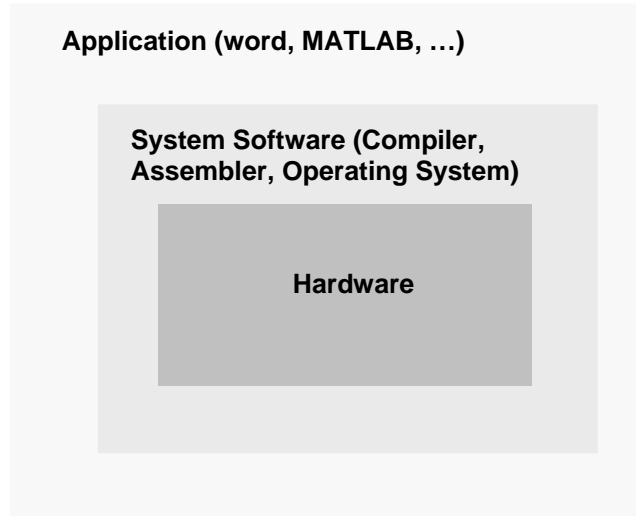
A typical desktop computer in 1985 had 512 Kbytes of RAM, where a typical desktop computer in 2008 has 2 GBytes ( $2 \times 10^9$  Bytes) of RAM. This represents a 4,000 fold increase in 23 years.

Student Exercise – Can you think of another industry or technology other than semiconductors that has shown similar growth?

### ❖ Computer Software Layers

Although there are a variety of computer designs and software, it can be said that computer software consists of the operating system and application software at the highest level. Below is a pictorial

view of the relationships between hardware and these two layers of software.

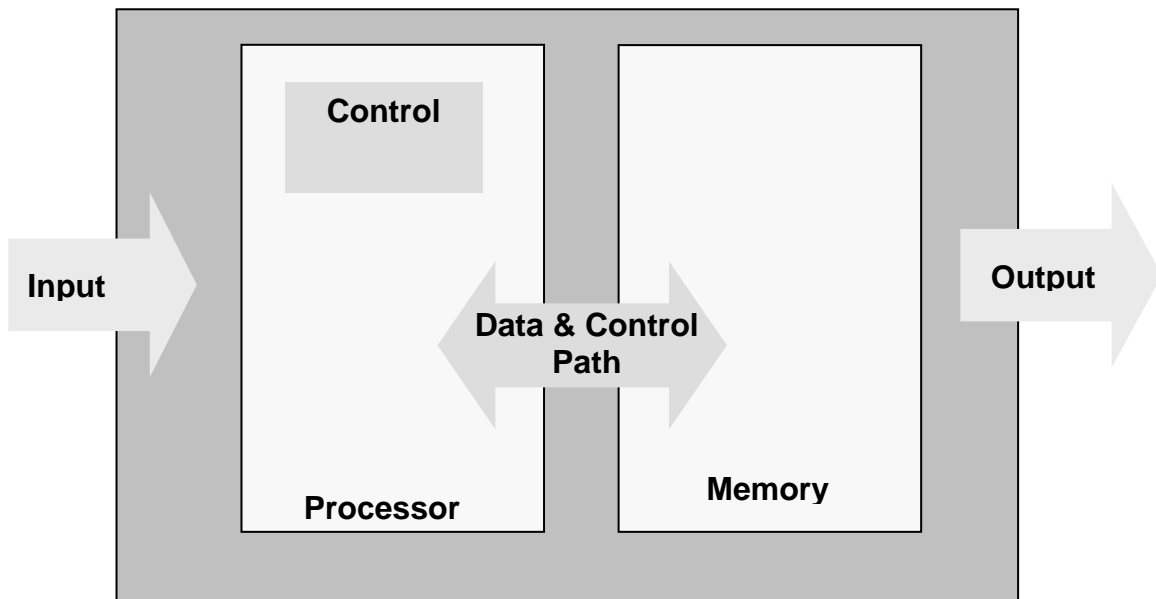


➤ **Operating System (OS)**  
Operating System software and its underlying support are responsible for controlling the functionality and resources of a computer on behalf of users and applications such as Microsoft Word or MATLAB. Microsoft Windows, Linux and MacOS are three of the most popular Operating Systems in the computer industry. A typical OS delivers the following services.

- Handles input/output
- Starts and stops application
- Allocates storage, memory, and processor, and in general, manages the resources among the applications (active processes).

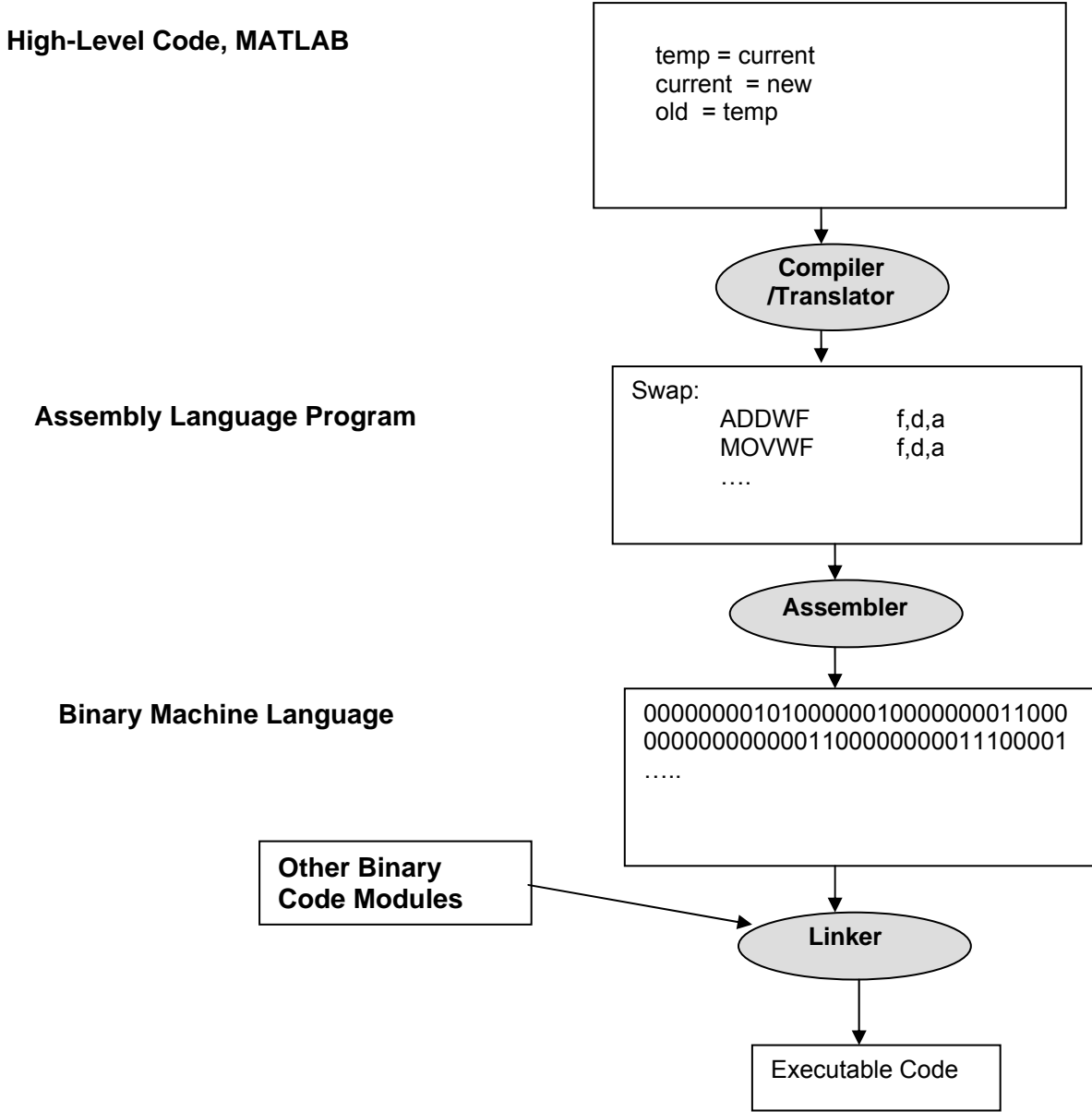
❖ **Computer Hardware Overview**

Computer Hardware has a processor build on silicone at its core with additional hardware for input, control, execution and output. The following Diagram shows the hardware architecture:



### 6.3. Programming Levels

Steps from High Level Language (C, Java, MATLAB...) to executable code includes compiling, Assembling and Linking as show below:



## 6.4. Software Development Steps

The initial tendency of a new software developer is to start typing their code as soon as possible. For most real problems, this approach will invariably take longer, and result in lower value solutions that have to be reworked. At the end it is hard to understand and maintain.

The early steps of software development have much higher impact on the effectiveness of the final solution than the coding task. Most businesses have discovered this concept and have their most skilled developers work on the initial phases of development, and less experienced developers work on the coding and testing.

Although the software development steps vary among organizations, depending upon their needs and the maturity of the programming staff, it is possible to organize the process of software development into four steps:

### ❖ Requirement Analysis

The "Requirement Analysis" is the most important step in the programming process. This is where the programming problem is defined. The more details that can be agreed to early on, the higher the probability of a successful project.

It is important for software developers to spend sufficient time and effort to fully understand the problem and customer needs. Additionally, the following points should be documented and agreed upon in this phase:

- Know the customer
- Itemize a list of functions that the program has to perform
- Reliability requirements
- Response requirements
- Usability and user interface requirements
- Estimated time to complete the design and implement the solutions
- List of unresolved issues and a plan to resolve them.

### ❖ Design Phase

There are many design processes and tools available on the market to document high-level design before moving on to the coding. Due to the importance of design, businesses are developing new tools and processes to improve design phase. In this Section, we discuss the use of Pseudo Code and Flow Chart as two of the most common design tools.

#### ➤ Pseudo Code

The most common method used is pseudo code. Pseudo code is a mix of English and programming language elements (with limited regard for correctness of syntax). The goal is to convey the architecture, major functional blocks and interfaces of your design to the reader and implementer. It also helps the designer to concentrate on the algorithms without the syntactic details. Note that the pseudo code cannot be compiled or executed.

Although we will not introduce specific rules for pseudo code, the designer should attempt to convey sufficient information so that the implementer can use the pseudo code to implement the final syntactically correct code. There is a fine balance between enough detail and clarity of the conceptual design.

- Example: Pseudo code for a program that counts from 0 to 10 and returns.

```
Count=0
```



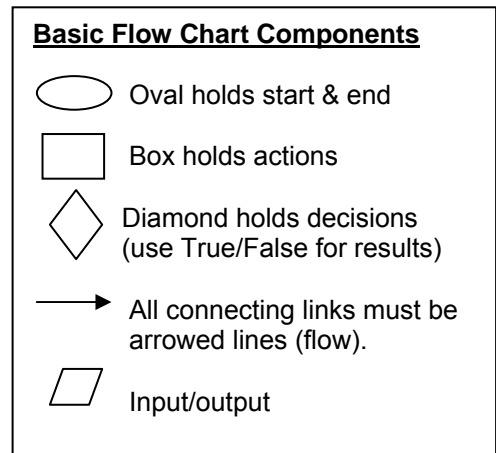
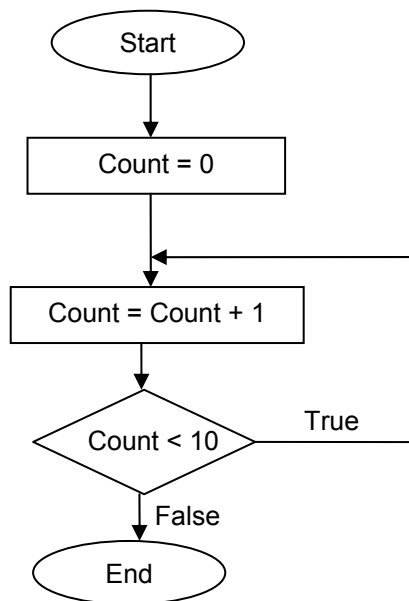
```
Count up to 10
Return
```

➤ Flow Chart

Flow charts allow for more detailed specifications and design description. It is also graphical and is easier to read and understand for the typical programmer and reviewer. The drawback is that flow charting for a complex project can become impractical due to the paper/computer screen size limitations. Many projects use flow charts for areas that require detailed definition while using the pseudo code for the rest of design.

There are a number of tools on the market for flow charting. One of the most popular flow charting tools is the Microsoft's Visio Software.

- Example: Write a program that counts from 0 to 10 and returns when done with the counting.



❖ Coding

Finally, this is the step in which we actually start to type commands into our development environment or write the program. Again, it is important to use the design definition as a guide for the coding. If a design or requirements error is found, it is important to update the earlier steps to correct the error before continuing with the coding task.

❖ Testing and Validation Step

As the name implies, the Testing and Validation Step ensures that the final program/code meets all the requirements agreed to at the start of the development process. If the requirements have been done with care and sufficient effort, then the testing would simply involve testing the program versus the requirements. Again, it is clear that investment in requirements pays dividends throughout the process.

## 6.5. Development Environment Interface and Structure

We will be using MATLAB as the Interactive Development Environment (IDE) and programming language in this text. MATLAB is an example of a high level language and is well-suited to introduce high level language. MATLAB is the most common language used in engineering and science education, as well as in industry to build models and solve engineering/scientific problems.

Today's engineering requires models to test concepts before committing to production or even development of prototypes. Science uses models to test theory and new concepts also. Additionally, MATLAB's C-like programming environment provides an excellent preparation for moving to advanced, high level programming and for other Engineering courses.

### ❖ MATLAB Overview (*extract from MATLAB Help Section*)

MATLAB system consists of five main parts:

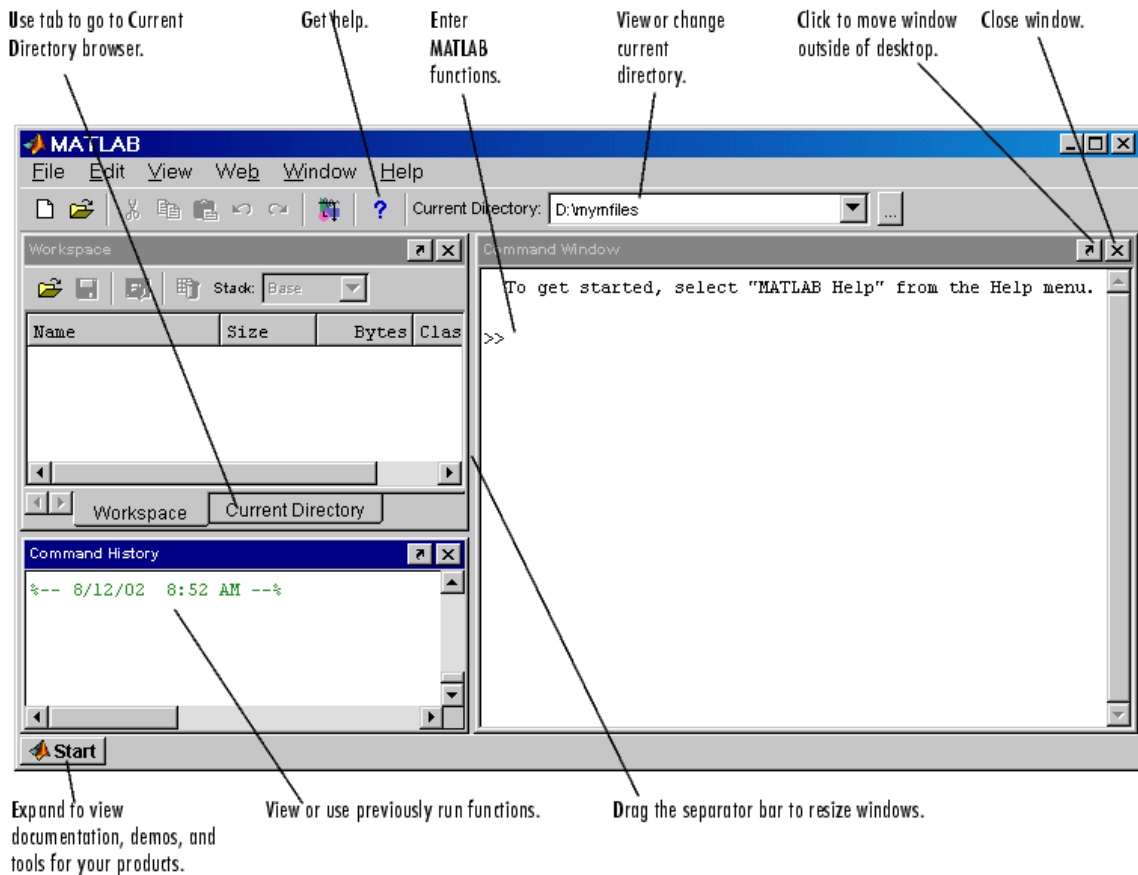
- Interactive Development Environment (IDE)  
The IDE includes the MATLAB desktop and command window, a command history, an editor and debugger, and browsers for viewing help, the workspace, files, and the search path.
- Mathematical Function Library  
A collection of computational algorithms ranging from functions like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.
- MATLAB Language  
A high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.
- Graphics  
MATLAB has extensive facilities for displaying vectors and matrices as graphs, as well as annotating and printing these graphs. It includes high-level functions for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level functions that allow you to fully customize the appearance of graphics as well as to build complete graphical user interfaces on your MATLAB applications.
- Application Program Interface (API)  
This library enables programs written in C and FORTRAN to interact with MATLAB.

### ❖ To start MATLAB, go MS Windows' Start menu and select the following:

<Start><MATLAB><MATLAB>

❖ Components of MATLAB Development Environment

The following window is the typical MATHLAB IDE start up window:

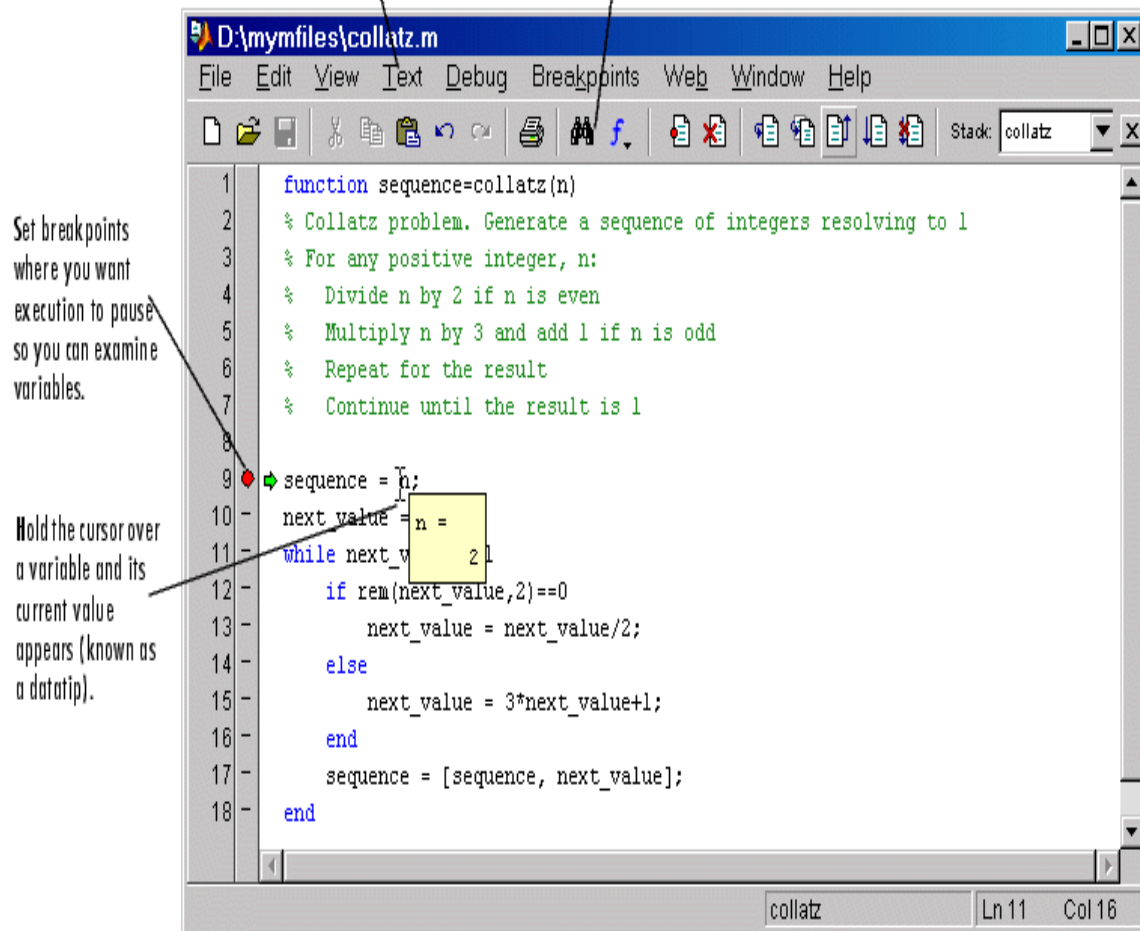


Developers have the ability to rearrange the components within the window to meet their programming needs and preferences. But it is recommended that the developers stay with the default configuration until they have completed a few programs and are familiar with MATLAB.

## ❖ Editor/Debugger Window

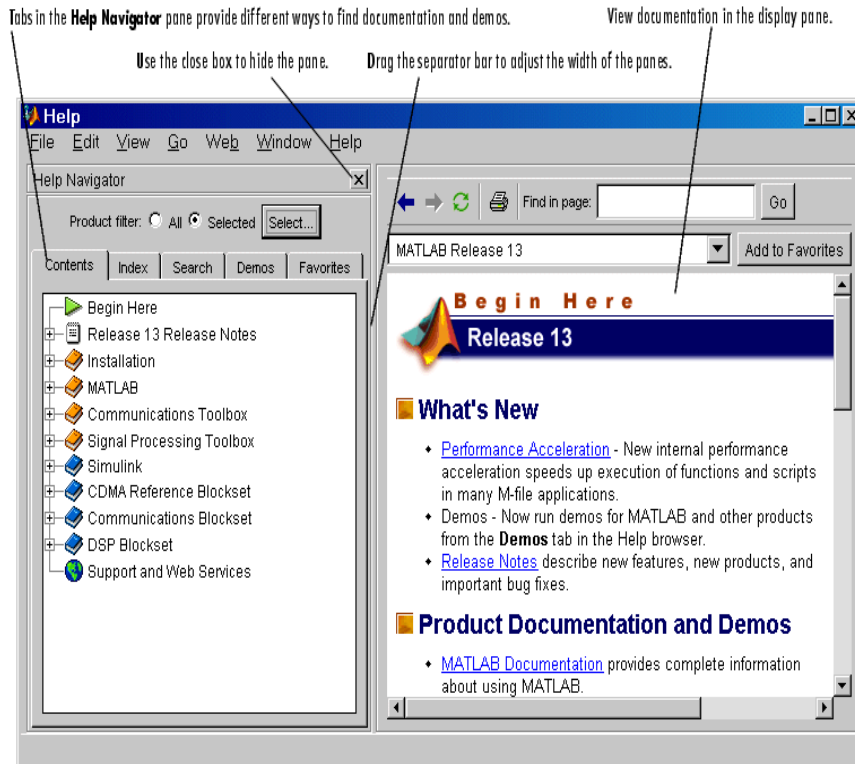
In the MATLAB Development Environment, right click in the Workspace window to create a new test m-file (test.m). Note that before a program can execute (run), it has to be typed in a file with extension “m” such as test.m. Once you have created a file, double clicking on the file name causes MATLAB to open the file in the Editor/Debugger window. The Editor/Debugger window allows execution of program and also viewing the results during the execution.

Comment selected lines and specify indenting style using the **Text** menu. Find and replace strings.



## ❖ Help Section

From the MATLAB Development Environment, select <help><MATLAB help> to see the following window.



MATLAB Help is the best resource available to learn about full functionality of MATLAB (Tutorial Section) or use the Search feature to look for a specific function.

A must for a new MATLAB Program is the demos in the Help Section. From MATLAB Help Window select the demo tab and review the various MATLAB demos to become familiar with MATLAB functionality.

It is important to become comfortable with the use of online documents to answer questions and find any required MATLAB information.

## 6.6. Using the MATLAB Command Window

The command window allows you to write equations and issue commands that will be executed by MATLAB. We will start by some useful basic commands that could be executed.

### ❖ Command window prompt

Each time you complete a command or when you start, ">>" indicates the location that the new command should be typed followed by the <enter> key.

A useful command is "clc" which clears the command screen. In the Command window, type "clc" followed by the <enter> key. When this command is executed, it will result in command screen being cleared. Type the command and try it!

```
>> clc
```

Note: The most effective approach to learning MATLAB and programming in general is by performing the tasks while you are reading about them.

### ❖ Help Command

Help command is one of the most useful commands since it provides you with brief usage information for any of MATLAB's available commands. For example, the following command will display a description of the clc command with links to any related commands and more reference material.

```
>> help clc
```

Use the Help menu items for complete descriptions of commands and their usages since Help command only provides a brief description as a refresher.

### ❖ Calculation

The Command Window can also be used to simply solve equations and do arithmetic, much like a calculator. For example, enter the following commands to make the calculation and save the results in x:

```
>> x=10 + 12* 20
```

By adding a ";" at the end of command line before pressing <enter> as shown below will cause the results of the calculation, to not be displayed:

```
>> x=10 + 12* 20;
```

The ";" is useful in programming when there is a long list of questions and only the final answer is of interest. By add ";" to the end of intermediate equations, only the final answer will be displayed.

### ❖ Case-Sensitive Variable

MATLAB is case-sensitive, which means if you assign a value to x, it will not be assigned to X. In order to experience the variable case-sensitivity in MATH, enter the following commands:

```
>> x = 10 + 5;  
>> x
```

At this point, the system shows display, the result 15. But now if you enter the following command:

```
>> X
```

The system will display “??? Undefined function or variable 'X'. message. Although x was equal to 15, no value has been assigned to X.

The Command Window allows you to enter individual commands, but it does not provide you with the ability to save multiple commands. The ability to save and rerun a set of commands (program) and share it with others is critical for programs with more than a few commands. Fortunately, MATLAB offers such a capability through M-files, which is covered in the next section.

## 6.7. Creating and Editing M-files

M-files allow the developer to write programs and save them for future use or later modification. It also opens the door to using functionality of a previously developed program in a new function.

### ❖ Creating an M-file

This process creates a file with extension “.m” which is referred to as an m-file. We can add a series of command in the m-file (for example refer to my\_add.m). Then the commands in the m-file can be executed from the command files. Start by selecting the following from the command window’s menu bar:

File > New > M-File or File > New > Function

This will open a new window, M-file Editor, which can be used to enter commands. But before we enter our desired command, there are some lines that must be added to the beginning and end of file. So first add the line “function [results] = my\_add (operand1, operand2). And add “end” after the last command in the file.

```
function [results] = my_add (operand1, operand2)
    results = operand1 + operand2;
end
```

The way we have define the above function, it will accept two parameters, “operand1” and “operand2”, adds them together, and places the results in the variable “results”.

In order to save the file from the M-file editor menu, select:

File > Save As

At this point, you can browse and select the specific directory you want your file saved in. It is a required practice to name the function the same as the file. In this case we will refer to it as my\_add.m. Take care to remember the file name and the saved directory, since you will need that information to run your MATLAB program.

Congratulations, you have created a MATLAB program!

Now, you should click on the Command window and type in the name of the file without the “.m” but make sure to add the operands. For example, if you would like to add the two numbers 6 and 10, type the following in the command window:

```
>> my_add(6, 10)
```

MATLAB displays “ans = 16”, which is the result of adding 6 and 10.

### NOTES:

- 1) MATLAB is case sensitive so “My\_add” is not the same as “my\_add”.
- 2) MATLAB does not allow the use of reserve words for variables, function names, or files names. The best practice is to customize the variables and names with the project specific designation to avoid potential conflicts.
- 3) Your current directory must be where the program is saved in order to execute it.

➤ Student Exercise - Write a program that accepts x as input and returns  $y=2x + 30$ .



Solution:

❖ Comments and Formatting Programs.

It is important and required in this course that programs contain information that helps the reviewer to understand the function of the program, the author, and the program revisions. Also, it is useful to add comments that describe complex functionality and assumptions that may not be obvious to the reviewer.

These comments are meant for the human reader and are not executed as part of the program. “%” informs MATLAB that the remainder of the line after “%” is comment for the human reader and not to be executed.

Here is a properly commented program:

```
% m-file creation example
% Author: Izad Khormaee
% Last update: 9/10/06, Version: 1.2
% Input parameters: operand1, operand 2
% output parameter: results
% my_add function adds the value of operand 1 and operand2; assigns it to the
% results variable.

function [results] = my_add (operand1, operand2)
    results = operand1 + operand2;
end
```

Also, it is important to use tabs (indentation) to show the statement blocks for ease of readability. For example, the statement “results = operand1 + operand2;” is indented to show that it is part of the function my\_add.

If the editor window is not open, use the command window menu “File > open” to open the m-file “my\_add.m” and make the changes to the file. Be sure to save the file before attempting to execute it from the command window.

Notes:

Placing “;” at end of a command prevents command results from being displayed during the program execution.

**Student Exercise** - Write a program that calculates the volume of a Sphere with Radius R which is provided by the user.

Hint:  $V = \frac{4\pi}{3} R^3$

## 6.8. MATLAB Arithmetic and Logic Operators

MATLAB supports arithmetic, algebra, trigonometry and other mathematical operations for real numbers, complex numbers, and matrices. We will discuss the complex numbers and matrices later. For now, it suffices to consider them as a more generalized form of real numbers.

MATLAB supports a variety of operators and the best source for learning about MATLAB operators is the Help Section of MATLAB. Here are some definitions extracted from MATLAB Help section for the most commonly used operators:

### ❖ Arithmetic Operators

Arithmetic operators detects the type of value being operated on and will perform the appropriate operation for the operator type.

- Addition “+”  
 $C=A+B$  → adds A and B, with the result being stored in C.
- Subtraction “-”  
 $C=A-B$  → subtracts B from A, with the result being stored in C.
- Multiplication “\*”  
 $C=A*B$  → multiplies A and B, with the result being stored in C.
- Division “/”  
 $C=A/B$  → divides A by B, with the result being stored in C.
- Power “^”  
 $C=A^B$  → A is raised to the power of B, with the result being stored in C.

### ❖ Logical Operators

The symbols “&”, “|”, and “~” are the logical operators corresponding to AND, OR, and NOT. The value of the results and operands may be logical 0, representing false, or logical 1 (may also be any nonzero value), representing true. The logical operators return 1 (true) or 0 (false), as appropriate.

Operator orders of operation are:

“~” the highest,  
“&”  
“|” the lowest.

- “~” operator – complement or NOT.  
 $C = \sim A$  is defined by the following truth table:

A	C
0	1
1	0

- “&” operator - Logical AND  
 $C = A \& B$  is defined by the following truth table:

A	B	C
0	0	0
0	1	0

1	0	0
1	1	1

- “|” operator - Logical OR  
 $C = A | B$  is defined by the following truth table:

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

#### ❖ Relational Operators

Relational Operators enable developers to compare two operands and take action based on the relative value of the two operands. Note that when the relationship is true, the result will be logical 1 (true); otherwise the result will be logical 0 (false).

Here are a selection of available relational operators:

- Less than operator “<”  
 $(A < B) \rightarrow$  True if A is less than B; otherwise false.
- Greater-than operator “>”  
 $(A > B) \rightarrow$  True if A is greater than B, otherwise false.
- Less-than-or-equal operator “<=”  
 $(A <= B) \rightarrow$  True if A is less than or equal to B; otherwise false.
- Greater-than-or-equal operator “>=”  
 $(A >= B) \rightarrow$  True if A is greater than or equal to B; otherwise false.
- Equal operator “==”  
 $(A == B) \rightarrow$  True if A is equal to B; otherwise false.
- Not-equal operator “!=”  
 $(A != B) \rightarrow$  True if A is not equal to B; otherwise false.

#### ❖ Examples

- Evaluate  $C = \sim 5 \& 4 | 0$

Solution:

$$C = (0 \& 1) | 0 = 0 \text{ or False}$$

- Evaluate  $C = A | B \& \sim A$  when A= False and B=True

Solution:

$$C = 0 | (1 \& 1) = 1 \text{ or True}$$

- Evaluate  $\{C == (A >= B) \& (B <= C)\}$  when A = C/B, B=2, C =5.5

Solution:

$$\{C == (2.75 >= 2) \& (2 <= 5.5)\}$$

```
{5.5== 1 & 1}  
{1}
```

**Student Exercise** - Write a program that accepts two inputs (in1, in2) and returns 1 if the two values are the same.

## 6.9. MATLAB Data Flow Controls

The program is executed sequentially by the computer unless a data flow control is used to redirect the program execution to a different location. Typically, data flow controls allow for a logical test using the relational operators to determine which statement should be executed.

The two most common data flow controls are the “for” loop statement to create a loop and the “if” conditional statement for decision making.

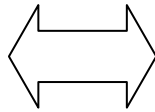
### ❖ “if” Conditional Statement

The statements following “if” will be executed if the conditional expression is true or non-zero. The ELSE and ELSEIF parts are optional; if you do not need them, then eliminate them. But as long as there is an opening “if”, there has to be a terminating “end”.

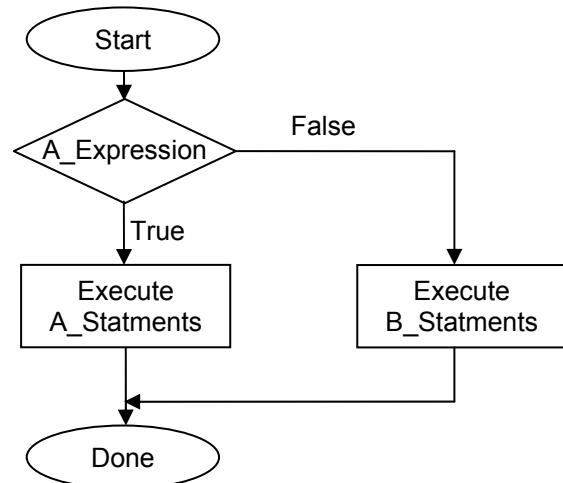
The following figure shows the syntax of a simple “if” conditional statement and the execution flow using a flow chart.

#### Simple “if” Statement Form

```
if A_expression
    A_statements
else
    B_statements
end
```



#### Flow Chart of “if” statement



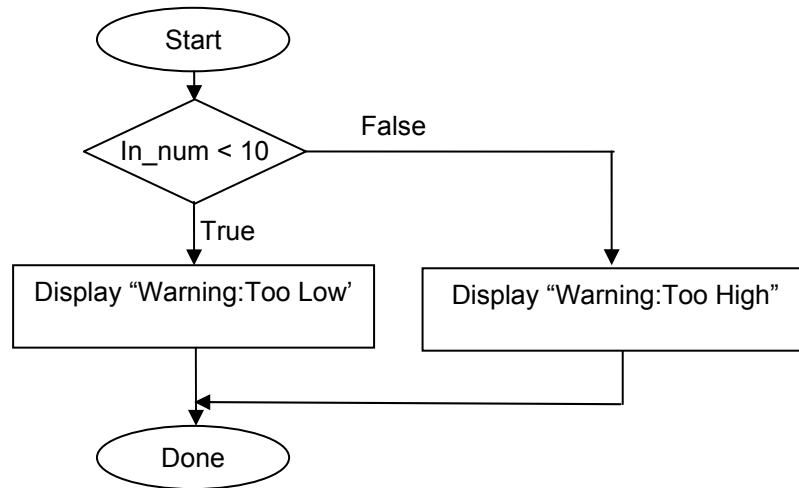
The A\_Expression is usually of the form:

(exp1 <operator> exp2)  
where operators are relational Operators such as ==, <, >, <=, >=, ~=.

For example (a < b) is a valid expression.

- Example – Write a program that accepts one input and when the input is less than 10, it displays “Warning: Too Low”. Otherwise display “Warning: Too High”.

Step 1) Translate Problem Statement to Flow Chart



Step 2) Translate the flow chart to code

```

% Example- Use of if statement

function [] = if_example(in_num)
    if (in_num < 10)
        disp ('Warning: Too Low');    % Displays the string
    else
        disp ('Warning: Too High');
    end
    % end if (in_num <10)
end
  
```

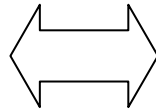
**Student Exercise** - Write a program that accepts two inputs; if the first input is larger than the second, the program displays the word "true", and otherwise it should display "false".

The following flow chart shows the more general form of “if” statement.

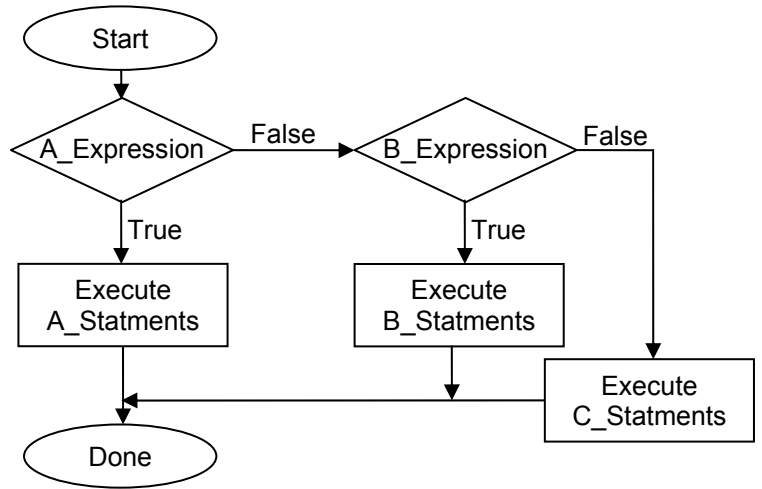
**General “if” Statement Form**

```

if A_expression
  A_statements
elseif B_expression
  B_statements
else
  C_statements
end
  
```

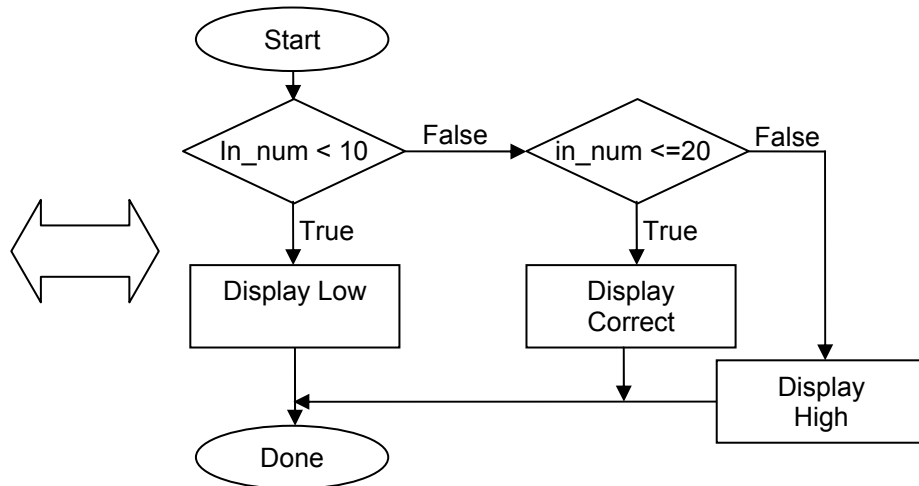


**Flow Chart for General “if” statement**



- Example – Write a program that accepts one input and if the input is less than 10, display “Warning: Too Low”. For an input between 10 and 20, displays “Correct Level”. Otherwise, it displays “Warning: Too High”. Each display should be followed by a beep.

Step 1) Translate problem statement to flow chart



Step 2) Translate flow chart to problem statement

```

% Example- Use of if statement

function [] = if_example(in_num)
  if (in_num < 10)
    disp ('Warning: Too Low');    % Displays the string
    beep;
  elseif (in_num <=20)
    disp ('Correct Level');
  end
end
  
```

```
        beep;
    else
        disp ('Warning: Too High');
        beep;
    end
                                % end if (in_num <10)

end
```

*Note: For code readability and maintenance, the code should be formatted using a tab indentation. The statements in the same scope (level) should have the same amount of indentation.*

**Student Exercise** - Rewrite the program from previous example, but only use the simple form of "if" statement (without "elseif").



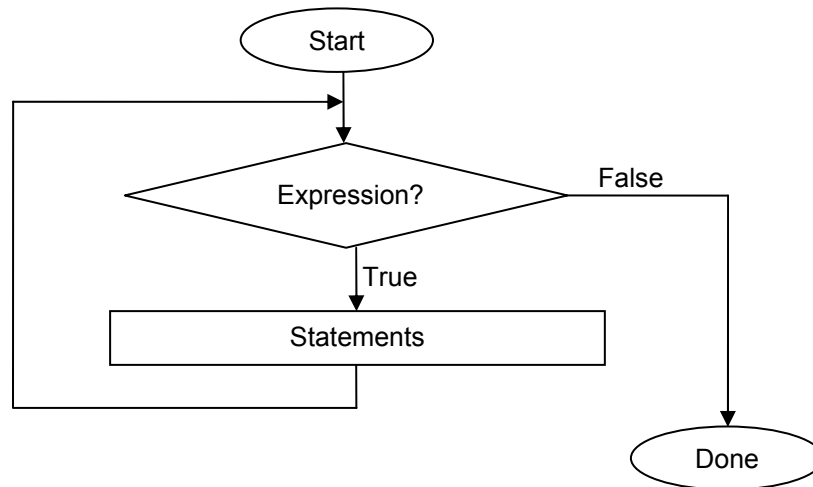
❖ “while” loop statement

The “while” loop construct repeats statements until the condition represented by the expression is not true.

```
while expression
  statements
end
```

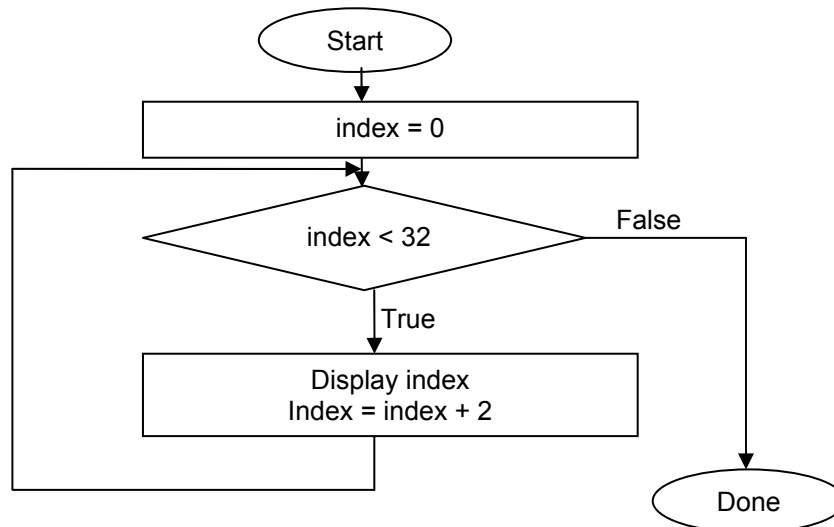
The statements are executed while the real part of the expression has all non-zero elements. The expression is in the form (expr1 rop expr2) where rop is ==, <, >, <=, >=, or ~=.

The following flow chart is a graphical representation of “while” loop statement:



➤ Example - Display all the even numbers from 0 to 31.

Step 1) Translate the problem statement to flow chart



Step 2) Translate the flow chart to code

```
% Example- Use of while Loop

function [ ] = while_example ()
    index = 0;
    while (index < 31)
        disp (index)    % displays index since we do not have “;” at the end of the line
        index = index + 2;
    end                % while Loop end statement
end
```

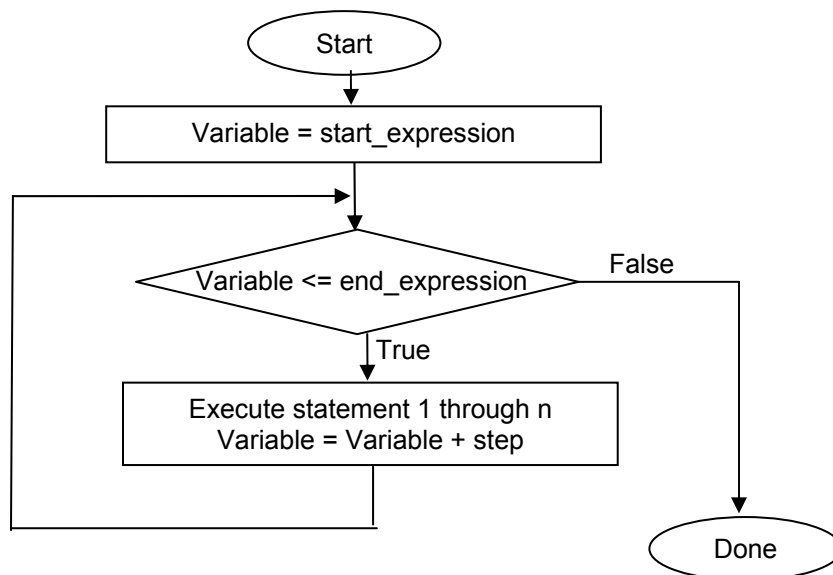
❖ “for” Loop Statement

The “for” Loop statement repeats a set of statements a pre-determined number of times. The general form of the “for” Loop statement is shown below:

```
for variable = start_expression: step : end_expression,
    statement1,
    ...
    statement n
end
```

This construct will execute statements 1 through n until the variable is out of the range of start\_expression to end\_expression.

The following flow chart graphically represents the function performed by the “for” loop:



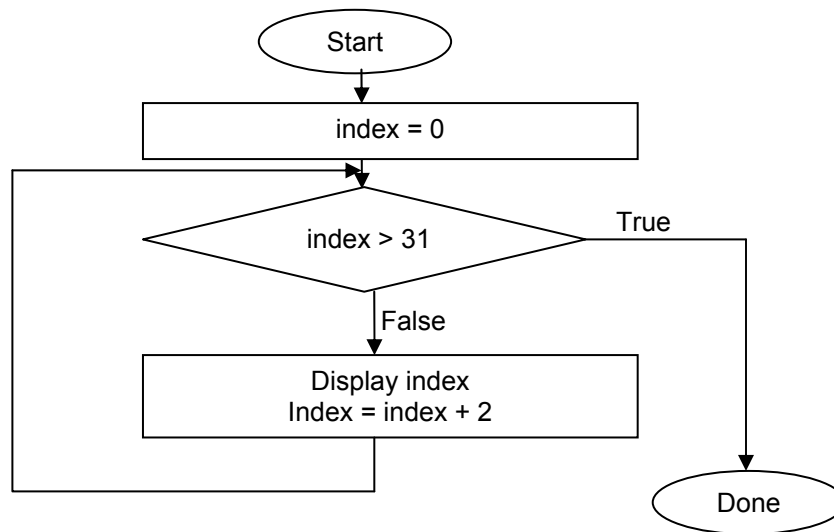
For example, the statement “for C=1:2:5” results in C taking on the value of 1 first time through the loop, value of 3 second time through the loop and value of 5 third time through the loop. It will not go through the loop a fourth time because C take on the value of 7 which is larger than 5 resulting in

termination of loop.

Finally, an important formatting consideration is indentation. By indenting the statement in a loop, the code becomes more readable, since it is visually obvious which statement is included in the loop and which is not.

- Example - Display all the even numbers from 0 to 31.

Step 1) Translate the problem statement to flow chart



Step 2) Translate the flow chart to code

```
% Example- Use of for Loop  
  
function [ ] = for_example ()  
    for index = 0:2:31,  
        index           % displays index since we do not have ";" at then of line  
    end                 % for Loop end statement  
end
```

Note : "for" Loops in its entirety is just a statement and as such it may be embedded in another "for" loops or other data flow structures.

**Student Exercise** - - Write a program that displays all odd numbers from 5 to 35.

Solution:

❖ Additional Common Functions

It is important to note that MATLAB has an extensive set of functionalities so use Help Section of MATLAB to search for new function. The following statement and functions are commonly used, so it may of use to the readers to review and understand their usage.

- “disp”  
The function disp(X) will display the content of X.

Example:

The following statements will display “this is a test”:

X='This is a test' // note that ' is used to define string value.

**disp** (X);

- “fprintf”  
The function fprintf(fid, format\_string, A, ...) formats the data in variable format\_string and display the information if fid is set to 1 or omitted.

The variable format\_string contains ordinary characters and conversion specifications.

Conversion specifications start with the character %,and conversion characters c, d, e, f, g, o, u, x and s. Below is the definition of each conversion characters:

Specifier	Description
%c	Single character
%d	Signed decimal notation
%e	Exponential notation (using a lowercase e as in 3.1415e+00)
%f	Fixed-point notation
%g	The more compact of %e or %f, are defined; the insignificant zeros do not print.
%o	Unsigned Octal notation
%s	String
%u	Unsigned decimal notation
%x	Hexadecimal notation

The special formats \n,\r,\t,\b,\f can be used to produce linefeed, carriage return, tab, backspace, and form feed characters respectively. Use \\ to produce a backslash character and %% to produce the percent character.

For example , The following code segment displays “This is 10 out of 23 pages”:

```
page_no = 10;
total_page = 23;
fprintf (1,'This is %d out of %d pages. \n', page_no, total_page);
```

- “mod”  
The function mod(x,y) calculates and returns the modulus or remainder after the division. After the execution of z=mod(x,y), the variable z will contain the remainder of x/y.

Example:

Statement “z=mod (125, 10)” results in value of z to be equal to 5 which is the remainder 125/10..

- “input”  
x = input ('Enter a value:') displays the string “Enter a value:” as a prompt for the use. Any

number entered will be stored in the variable x.

➤ “tic” and “toc”

Functions “tic” and “toc” are commonly used to time the execution of a program. “tic” starts the stopwatch timer and save the current time. “toc: measures the elapse time. So to time your operation use “tic” and “toc” as shown below:

```
TIC
Operation // code to be timed
TOC
```

The above code display the time used to complete the operation specified here by "operation" in seconds. Other related functions are “cputime” , “clock” and “etime”.

➤ More...

The functions presented here are a small introductory set with limited explanation. MATLAB has an extensive set of functionalities and ability to use matrices as variables and parameters. More information on presented functions and others are available on the Help Section of MATLAB.

To access the Help Section, from the MATLAB menu select:

```
Help > MATLAB help
Click on Search tab
```

❖ Additional Examples

- Write a program that calculates the number of transistors if each transistor takes up 30 nm<sup>2</sup> on a device. User will input the width and length of available silicon area of the device.

Solutions: Student Exercise

- Write a program that allows the user to input current (in Amp) and voltage (in Volts). With each voltage and current entry, your program calculates the remaining power from a 3 MW power source.

Solutions: Student Exercise

- Draw a flow chart and write a program that calculates the next 20 Fibonacci numbers following the sequence {1, 2, 3, 5, 8, 13, 21}.

Solutions: Student Exercise

## 6.10. Additional Resources

- ❖ MathWorks. [MATLAB Reference Material Version R2000a](#). (2007) MathWorks

## 6.11. Problems

Instructions:

- Matlab or SciLab may only be used for program development
- Homework solution should include flow chart and source code
- Be prepared to present your programs to the class

---

1. List the steps from High level language to executable computer code.

---

2. Write a program that displays the first 20 even numbers beginning with 100.

---

3. Write a program that accepts three parameters “lower\_limit”, “upper\_limit” and “factor”. The program should display all the numbers between “lower\_limit” and “upper\_limit” that are evenly divisible by “factor”.

---

4. Write a program that beeps every time the input parameter is evenly divisible by 13.

*Hint: beep() function produces sound as long the speaker is not on mute.*

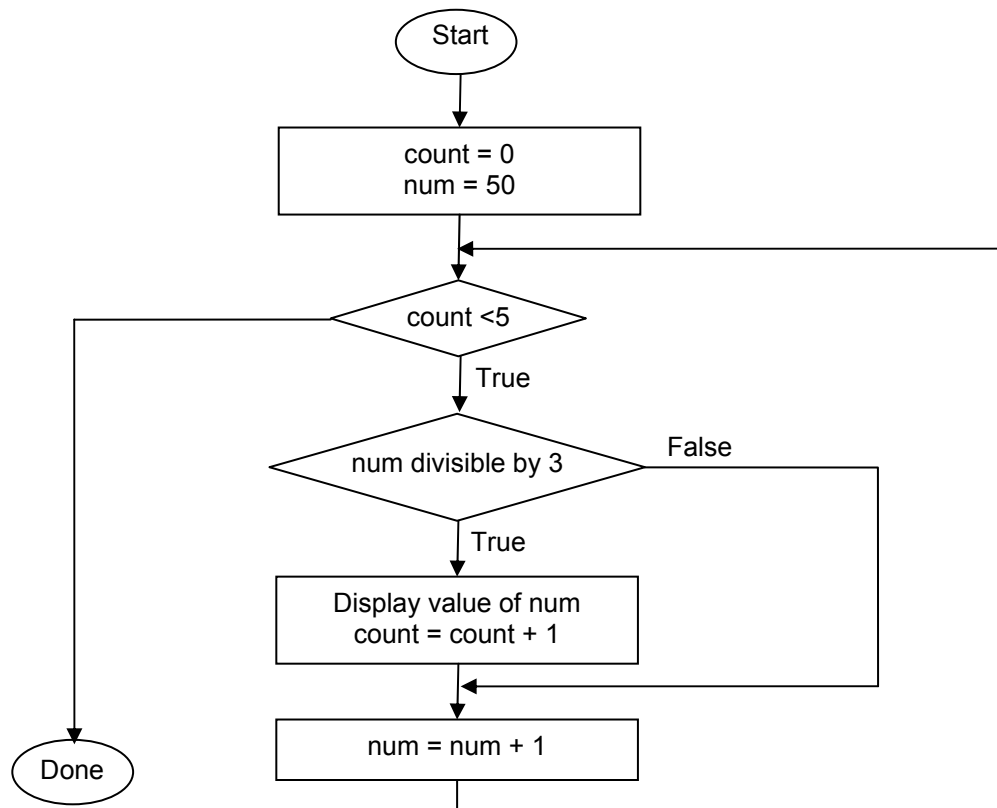
---

5. Write a program that displays the 50 consecutive prime numbers starting with 2. Use MATLAB to display the CPU time required to execute this program.

*Hint: tic() and toc() functions may be used. Standard Matlab functions such as prime() may not be used.*

---

6. Write a program that implements the following flow chart:



---

7. Plot and label the following functions for t from 1 to 50 in 0.1 steps:

a)  $x(t) = t^2 + 2t + 30$

b)  $y(t) = t^9 - 2t^6 + 7t^2 - 100t + 30$

---

8. Write a program that accepts a real number and displays the number in scientific format "x.xxxEn". For example 2546 would be displayed as 2.546000e+003

---

9. Write a program that accepts a fraction as two parameters "in\_numerator" and "in\_denominator". The program should remove all common factors between the two parameters and display "out\_numerator / out\_denominator" such that the value of fraction is maintained but there are no common factors between the "out\_numerator" and "out\_denominator".

---

10. Draw a flow chart and write the code for counting how many numbers from 10 to 149 is evenly divisible by 3.

---

11. Write a program that displays all numbers between input parameters "lower\_limit" and "upper\_limit" that have at least one digit equal to '2'.

*Note: lower\_limit and upper\_limit are positive integers.*



## **Chapter 7. Mathematical Concepts**

This Chapter discusses a number of key mathematical concepts with significant applications to engineering. The concepts covered include introductory treatment of trigonometry, complex numbers, and matrices. Further, the focus of this Chapter is on the application of these mathematical concepts to ECS fields.

### **7.1. Key Concepts and Overview**

- ❖ Matrices
- ❖ MATLAB Matrix Operations
- ❖ Trigonometry
- ❖ MATLAB Trigonometry Operations
- ❖ Complex Numbers
- ❖ MATLAB Complex Number Operations

## 7.2. Matrices

Matrices are another important concept when the system has multiple input, output or parameters. This type of system is typical of most realistic systems. The terms “matrix” and “array” are used interchangeably. Matrices enable us to combine multiple inputs or output parameters into a single variable. Use of matrices reduces the complexity of solving problems with multiple equations and unknowns which are common in engineering.

### ❖ Matrix Format

A matrix is written with its elements in rows and columns. Each matrix element is referred to by its corresponding row and column. Below is the general form of a matrix:

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{pmatrix}$$

n columns  $\longrightarrow$  m rows  $\downarrow$

More definitions:

- The above matrix is said to be an “m x n matrix”, where “m” refers to the number of rows and “n” refers to the number of columns.
- In the general expression “ $a_{k,l}$ ”, “k” refers to the row and “j” refers to the column corresponding to the location of the element in the matrix.
- “A” is called a row vector when m is equal to 1.

$$A = \left( a_{1,1} \quad a_{1,2} \quad \dots \quad a_{1,n} \right)$$

- “A” is called a column vector when n is equal to 1.

$$A = \begin{pmatrix} a_{1,1} \\ a_{2,1} \\ \vdots \\ a_{m,1} \end{pmatrix}$$

### ❖ Matrix Arithmetic

#### ➤ Adding Matrices

Matrices can only be added if the matrices have exactly the same dimensions. The resulting matrix's elements are produced by the addition of the corresponding elements from the operand matrices. Elements of Matrix C which are the result of the addition of matrices A and B can be calculated as shown below:

$$c_{j,k} = a_{j,k} + b_{j,k}$$

- Example – Adding two 2x3 matrices.

$$\begin{pmatrix} 1 & 3 & 5 \\ 2 & 4 & 7 \end{pmatrix} + \begin{pmatrix} 6 & 10 & 8 \\ 9 & 12 & 11 \end{pmatrix} = \begin{pmatrix} 6+1 & 10+3 & 8+5 \\ 9+2 & 12+4 & 11+7 \end{pmatrix} = \begin{pmatrix} 7 & 13 & 13 \\ 11 & 16 & 18 \end{pmatrix}$$

➤ **Subtracting Matrices**

Subtracting has the same requirements and process as addition. Elements of Matrix C which are the result of Subtraction of matrices A and B can be calculated as shown below:

$$c_{j,k} = a_{j,k} - b_{j,k}$$

- Example – Subtracting two 2x3 matrices.

$$\begin{pmatrix} 10 & 13 & 5 \\ 2 & 14 & 7 \end{pmatrix} - \begin{pmatrix} 6 & 23 & 8 \\ 9 & 4 & 11 \end{pmatrix} = \begin{pmatrix} 10-6 & 13-23 & 5-8 \\ 2-9 & 14-4 & 7-11 \end{pmatrix} = \begin{pmatrix} 4 & -10 & -3 \\ -7 & 10 & -4 \end{pmatrix}$$

➤ **Scalar Multiplication**

Scalar refers to a single value. In the Scalar Multiplication, the scalar value is multiplied by all the elements in the matrix. So the scalar value “a” is multiplied by matrix B, resulting in a matrix C where each element is calculated by:

$$c_{j,k} = a * b_{j,k}$$

- Example – Scalar (4) multiplied with 2x2 Matrix

$$4 * \begin{pmatrix} 6 & 23 \\ 9 & 4 \end{pmatrix} = \begin{pmatrix} 4 * 6 & 4 * 23 \\ 4 * 9 & 4 * 4 \end{pmatrix} = \begin{pmatrix} 24 & 92 \\ 36 & 16 \end{pmatrix}$$

➤ **Matrix Multiplication**

Multiplication of two matrices requires that the number of columns of the first matrix be equal to the number of rows in the second matrix. Note that order of multiplication is important, meaning A\*B is not necessarily equal B\*A and not all matrices can be multiplied.

Elements of C, the resulting matrix from multiplication of two matrices A\*B where a pxn matrix and B is nxq matrix, can be calculated using:

$$c_{i,j} = \sum_{k=1}^n a_{i,k} * b_{k,j}$$

It is also important to note that the resulting matrix will have pxq dimension.

- Example: Multiplication of 3x2 matrix by 2x3 matrix

Note: The number of columns of first matrix and the number of rows of the second matrix are the same (2); therefore, the resulting matrix will be a square matrix 3x3.

$$\begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix} * \begin{pmatrix} 10 & 30 & 50 \\ 20 & 40 & 60 \end{pmatrix} = \begin{pmatrix} (1*10 + 4*20) & (1*30 + 4*40) & (1*50 + 4*60) \\ (2*10 + 5*20) & (2*30 + 5*40) & (2*50 + 5*60) \\ (3*10 + 6*20) & (3*30 + 6*40) & (3*50 + 6*60) \end{pmatrix}$$

$$= \begin{pmatrix} 90 & 190 & 290 \\ 120 & 260 & 400 \\ 150 & 330 & 510 \end{pmatrix}$$

❖ Matrix Properties

➤ Associativity

$$A*(B*C) = (A*B)*C$$

➤ Distributivity

$$A*(B+C) = A*B + A*C$$

### 7.3. MATLAB MATRIX OPERATIONS

MATLAB offers a complete set of matrix operations. In addition to the operations already discussed, such as add, multiply and divide, the following functions are worthy of consideration:

❖ Entering a matrix

A matrix can be entered into MATLAB by entering each row element separated by a comma “,” and each row is separated by semicolon “;”.

To enter  $A = \begin{pmatrix} 6 & 8 \\ 9 & 4 \end{pmatrix}$ , type in the following command:

```
>> A=[6, 8; 9,4]
```

➤ Matrix with rows and columns of zeros

The command `zeros(r,c)` creates a matrix that contains r rows and c columns of 0s.

To create a 3x4 matrix of zeros:  $A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

type in the following command:

```
>> A=zeros(3, 4)
```

➤ Matrix with rows and columns of ones.

The command “`ones(r,c)`” creates a matrix that contains r rows and c columns of 1s.

To create a 3x2 matrix of ones:  $A = \begin{pmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}$

Type in the following command:

```
>> A=ones(3, 2)
```

➤ Custom Matrix Definition

MATLAB allows the definition of a matrix by entering values or using the “zeros” and “ones” commands.

For example, to create the matrix  $A = \begin{pmatrix} 2 & 3 & 4 & 5 \\ 1 & 1 & 1 & 1 \\ 9 & 9 & 9 & 9 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

Type in the following command

```
>> A=[2, 3, 4, 5; ones(1,4); 9*ones(1,4); zeros(1,4)]
```

➤ Matrix creation by indexing

MATLAB offers the ability to create a matrix by indexing. The following MATLAB command is used to create a matrix by indexing.

```
A = [N : K : M]           % Matrix A starts with N and increments by K each time
                                % This process continues as long as the value is less than or equal to M.
```

For example, the command  
 >> A=[3 : 2 : 9]

Returns the following matrix  
 [3, 5, 7, 9]

❖ Accessing elements of a matrix.

Matrix elements are identified by their location in the matrix in terms of their row and column number. It is important to note that row and column numbers start with 1 and are always positive integers.

Each element of a matrix can be accessed by using the name of the array and the row and column number. For example in the following array:

$$A = \begin{pmatrix} 11 & 12 & 13 & 14 \\ 21 & 22 & 23 & 24 \\ 31 & 32 & 33 & 34 \\ 41 & 42 & 43 & 44 \end{pmatrix}$$

element in row 3 and column 2 is access by the following command:  
 >> A(3,2)

MATLAB also allows access to a range of elements in a matrix. For example, to access the elements from row 2 to 3 and column 1 to 4, use the following command:  
 >> A(2:3, 1:4)

The above command returns a 2x4 matrix that is shown below:

$$\begin{pmatrix} 21 & 22 & 23 & 24 \\ 31 & 32 & 33 & 34 \end{pmatrix}$$

❖ Term-by-term operators

Term-by-term operators are “.\*” for multiplication, “./” for division, and “.^” for power. These operator perform the operation on each element and saves the result in the corresponding elements of the results matrix.

➤ Matrix term-by-term Multiply Example

$$A = \begin{pmatrix} 6 & 8 \\ 9 & 4 \end{pmatrix}, B = \begin{pmatrix} 1 & 3 \\ 2 & 5 \end{pmatrix}$$

$$\text{Use MATLAB to Evaluate } C = A .* B = \begin{pmatrix} 6*1 & 8*3 \\ 9*2 & 4*5 \end{pmatrix} = \begin{pmatrix} 6 & 24 \\ 18 & 20 \end{pmatrix}$$

❖ System of equations

One of the main applications of matrices is their use in solving a system of equations. By writing the equations using the matrix concept, we are able to use MATLAB.

For example, a system of equations can be transformed as shown below:

$$\begin{cases} 2x + 3y + 8z + 7w = 1 \\ 3x - 8y + 2z + 9w = -3 \\ 5x + 4y - 4z + 5w = 5 \\ x + 3y - 6z + 8w = 10 \end{cases} \Rightarrow \begin{pmatrix} 2 & 3 & 8 & 7 \\ 3 & -8 & 2 & 9 \\ 5 & 4 & -4 & 5 \\ 1 & 3 & -6 & 8 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = \begin{pmatrix} 1 \\ -3 \\ 5 \\ 10 \end{pmatrix}$$

Now if we make the following assignments:

$$A = \begin{pmatrix} 2 & 3 & 8 & 7 \\ 3 & -8 & 2 & 9 \\ 5 & 4 & -4 & 5 \\ 1 & 3 & -6 & 8 \end{pmatrix}, \quad X = \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix}, \quad C = \begin{pmatrix} 1 \\ -3 \\ 5 \\ 10 \end{pmatrix}$$

The original system of equations can be rewritten as:

$$A \cdot X = C$$

Multiply both sides with the inverse of A  $\rightarrow A^{-1}$

$$X = A^{-1} \cdot C$$

By writing a program in MATLAB that describes the above equation, we will be able to solve for the unknown X. Here are the commands

```
C=[1;-3;5;10];
A=[2,3,8,7;3,-8,2,9;5,4,-4,5;1,3,-6,8];
X=A^(-1)*C
```

which return the value of the unknown X:

$$X = \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} = \begin{pmatrix} -0.67 \\ 0.72 \\ -0.55 \\ 0.65 \end{pmatrix} \rightarrow x=-0.67, y=0.72, z=-0.55, w=0.65$$

**Student Exercise** - Using MATLAB, solve the following system of equations:

$$\begin{cases} 2x + 20y + 22z + 95w + 65t = 2 \\ 31x + 21y - 20z + 15w + 2t = -5 \\ 15x - 14y + 29z - 18w + 13t = 3 \\ 25x + 18y + 24z - 12w + 16t = 9 \\ 17x + 14y + 21z - 10w - 5t = 12 \end{cases}$$

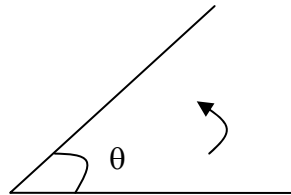
Solution:

## 7.4. Trigonometry

Trigonometry is a branch of mathematics that deals with relationships between distance and angles. This section discusses the phase or angle, followed by the four main trigonometric functions: sine, cosine, and tangent and cotangent.

### ❖ Angle or Phase

An angle or phase measures the rotation of a line compared to another as shown by  $\theta$  in the following diagram



### ➤ Measurements

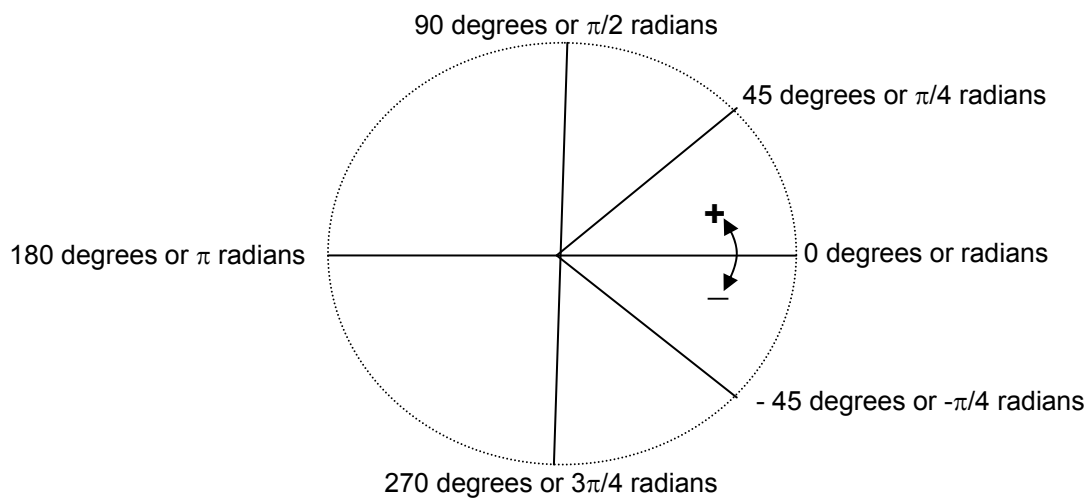
Angle measurements can be from 0 to 360 degrees (full rotation counter-clockwise around the circle). A degree can also be represented by superscript "0". For example, 35 degrees may be shown as  $35^\circ$ .

Another angle measurement is the radian. Angle measures in radian go from 0 to  $2\pi$  radians for full counter-clockwise rotation around the circle.

Degrees and radians may be converted using the following relationships:

$$\text{Radian} = \left(\frac{2\pi}{360}\right)\text{Degree} \text{ or } \text{Degree} = \left(\frac{360}{2\pi}\right)\text{Radian}$$

Refer to the following diagram for examples of various angles:



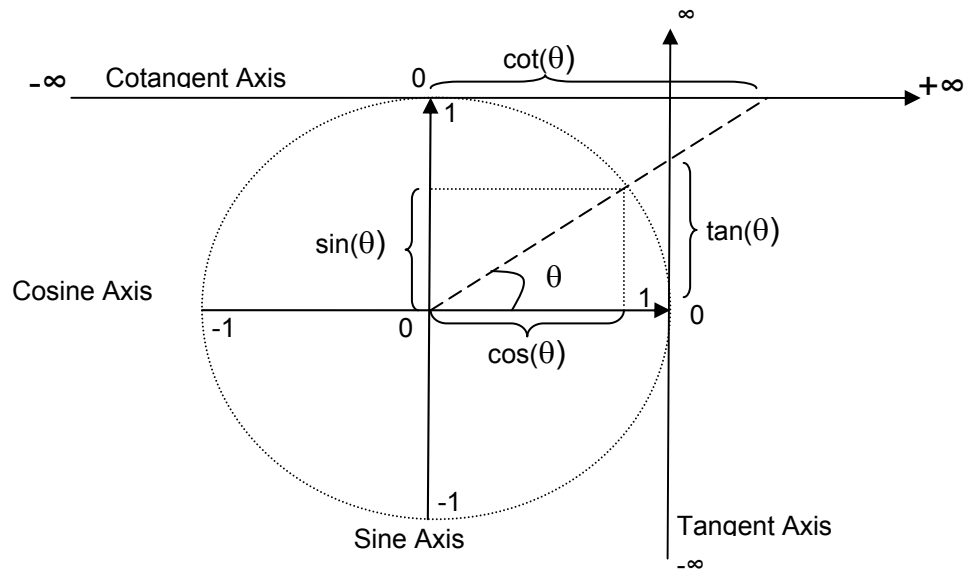


As shown in the diagram, counter-clockwise rotation is positive and clockwise rotation is negative.

❖ **Trigonometry Functions**

There are four trigonometry functions (Sine, Cosine, Tangent and Cotangent) that are commonly used in engineering and science fields. Each of these functions shows a relationship between angles and sides of a right-angle triangle.

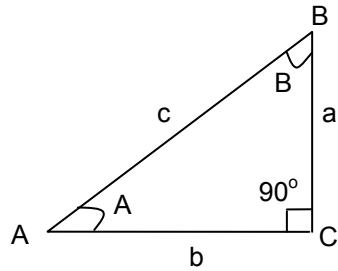
Typically, a Unit Circle (Circle with radius equal to 1) is used to show the values of each of the four trigonometry functions.



As it can be seen from the Unit Circle presentation of a trig function, each function has specific upper and lower limits as shown in the following table:

	Maximum	Minimum
$\sin(\theta)$	1	-1
$\cos(\theta)$	1	-1
$\tan(\theta)$	$\infty$	$-\infty$
$\cot(\theta)$	$\infty$	$-\infty$

The next step is to define each of the four trigonometry functions in terms of the associated sides of a right angle triangle.



$$\sin(A) = \frac{\textit{Opposite}}{\textit{Hypotenuse}} = \frac{a}{c}$$

$$\cos(A) = \frac{\textit{Adjacent}}{\textit{Hypotenuse}} = \frac{b}{c}$$

$$\tan(A) = \frac{\textit{Opposite}}{\textit{Adjacent}} = \frac{a}{b}$$

$$\cot(A) = \frac{\textit{Adjacent}}{\textit{Opposite}} = \frac{b}{a}$$

➤ Sine and Cosine Relationships

- Pythagorean Identity

$$\textit{Pythagorean Theorem} \Rightarrow a^2 + b^2 = c^2$$

$$\textit{Divide by } c^2 \Rightarrow a^2 / c^2 + b^2 / c^2 = c^2 / c^2$$

$$\textit{Substitute sin \& cos} \Rightarrow \sin^2 \theta + \cos^2 \theta = 1 \quad \text{"Pythagorean Identity"}$$

- Side angles

$$\sin(A) = \cos(B) = \frac{a}{c}$$

$$\cos(A) = \sin(90^\circ - A)$$

➤ Tan and Cot Relationships

$$\tan(A) = \frac{\sin(A)}{\cos(A)}$$

$$\cot(A) = \frac{\cos(A)}{\sin(A)}$$

$$\cot(A) = \frac{1}{\tan(A)}$$

➤ Other Useful Relationships

$$\sin(A \pm B) = \sin(A)\cos(B) \pm \cos(A)\sin(B)$$

$$\cos(A \pm B) = \cos(A)\cos(B) \mp \sin(A)\sin(B)$$

➤ Power Series

Computers use the power-series approach to calculate trigonometry functions. These series were developed by Newton and other mathematicians in the 17<sup>th</sup> century.

Here are the power series for angle x (where x is in radians):

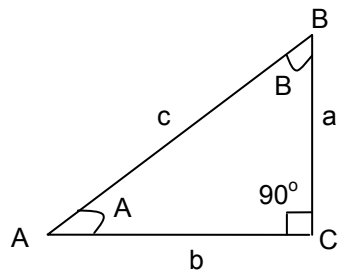
$$\sin(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{(2n+1)}}{(2n+1)!} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots$$

$$\cos(x) = \sum_{n=0}^{\infty} (-1)^n \frac{x^{2n}}{2n!} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots$$

➤ Inverse Functions

As can be seen from the relationships, any one of the trig functions can be derived from any other trig function.

Additionally, each function has an inverse where if the value of trig function was known then the inverse function provides us with the angle. The inverse function either use superscript “<sup>-1</sup>” or arc in front of the function to indicate that it is the inverse.



$$\arcsin\left(\frac{a}{c}\right) = \sin^{-1}\left(\frac{a}{c}\right) = A$$

$$\arccos\left(\frac{b}{c}\right) = \cos^{-1}\left(\frac{b}{c}\right) = A$$

$$\arctan\left(\frac{a}{b}\right) = \tan^{-1}\left(\frac{a}{b}\right) = A$$

$$\text{arccot}\left(\frac{b}{a}\right) = \cot^{-1}\left(\frac{b}{a}\right) = A$$

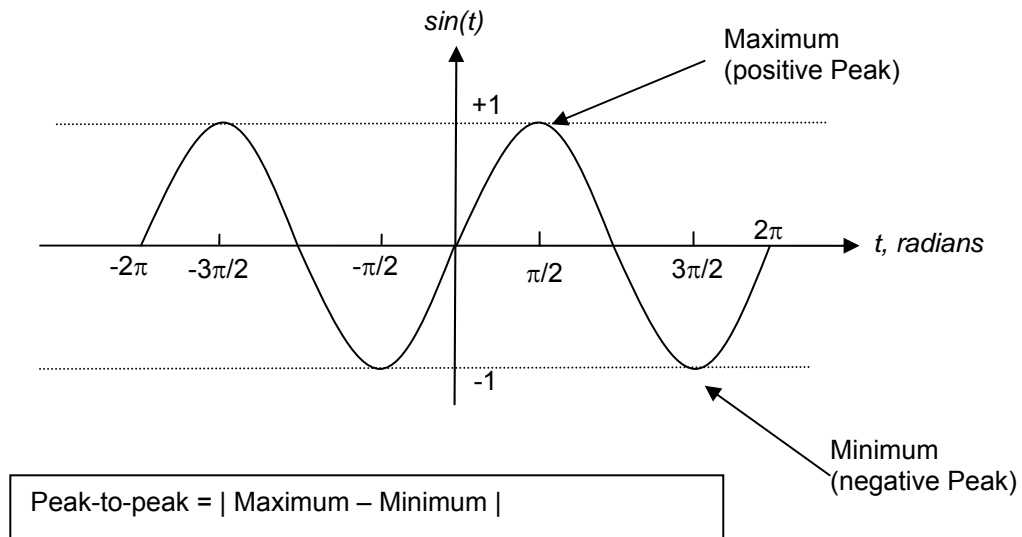
❖ Graphing and Application

When the value of an angle changes then the value of sine changes correspondingly. Electrical signals, sound waves and even light waves travel in time corresponding with the shape of a sine wave (sinusoidal signal).

Below are the graphs of a sine wave when angle is equal to “t”, “wt” and “wt + φ”.

➤ Sine graph when angle is “t”

In this case, let t be in radians. We can plot sin(t) as shown below:



- Sine graph when angle is “wt”  
In this case, we are redefining the angle with parameters that are commonly used to describe electrical, audio and light waves.

The following is plot of  $\sin(\omega t)$  where:

$t$ , represents time in seconds

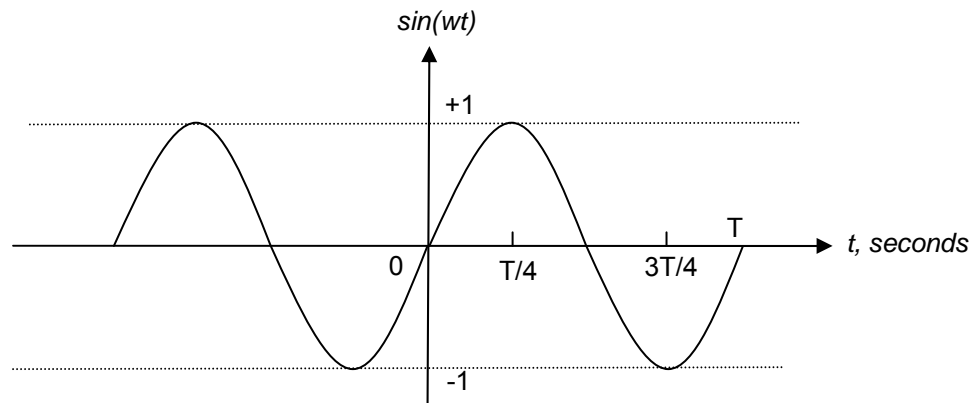
$\omega$ , represents angular frequency in radians/second

$\omega = 2\pi f = 2\pi/T$  where:

$f$ , represents the frequency in Hertz or cycles/second

$T$ , represents the period in seconds

A period is the time required to complete one of the repeating patterns.



- AC Electrical Signal coming to homes in the United States  
When viewing the signal voltage on the oscilloscope, the signal observed is similar to the one shown above. The frequency of AC signal that comes to homes is 60 hertz ( $f=60$ ).

We can use the  $f=1/T$  relationship to calculate the signal period of  $T=1/60$  seconds.

- Phase Shifting - Shifting the sine wave over time  
In order to shift the sine wave over time, we simply can add a positive or negative constant (phase shift) to move it to the left or right correspondingly. The constant  $\phi$  in  $(wt \pm \phi)$  represents the phase shift.

The following is the plot of  $\sin(\omega t - \phi)$  where:

$t$ , represents time in seconds

$\omega$ , represents angular frequency in radians/second

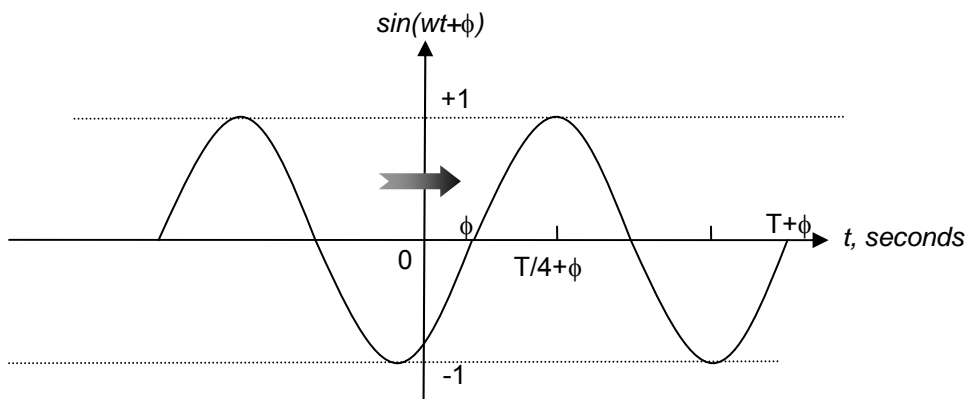
$\omega = 2\pi f = 2\pi/T$  where:

$f$ , represents the frequency in Hertz or cycles/second

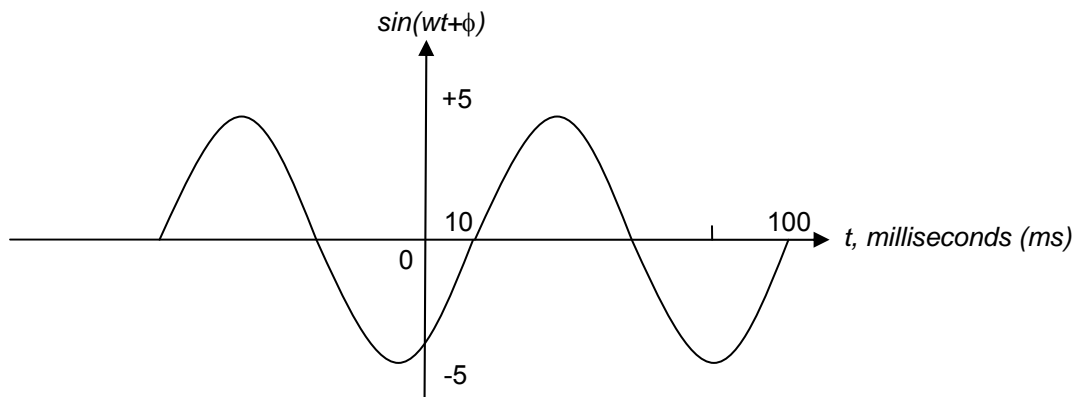
$T$ , represents the period in seconds

A period is the time required to complete one of the repeating patterns.

$\phi$ , phase shift in radians.



- ❖ Example – Write the function  $f(t)$  that is represented by the following diagram



Solutions: Student Exercise

## 7.5. MATLAB Trigonometry Operations

MATLAB enables engineers and scientists to visualize and model problems, systems and solutions. Sinusoidal signals play an important role in modeling and analysis of engineering and scientific systems. MATLAB's trigonometry functions and its graphing capabilities are useful in analyzing/designing time- and frequency-dependent sinusoidal-based systems.

### ❖ Calculating Trigonometry Functions

By default, MATLAB assumes that angles are in radians ( $\pi$  radians = 180 degrees), so remember to convert all angles to radians. Another useful fact is that MATLAB uses the function `pi()` to return the value of  $\pi$ .

MATLAB can calculate all the trigonometry functions. Below are the most useful ones:

- Sine,  $y=\sin(x)$ , and Inverse Sine,  $x=\text{asin}(y)$   
Examples:
  - (a) `sin(pi/4)` returns 0.707
  - (b) `asin(0.707)` returns  $0.7852 = (3.14)/4 = \pi/4$
- Cosine,  $y=\cos(x)$ , and Inverse Cosine,  $x=\text{acos}(y)$   
Examples:
  - (a) `cos(pi/3)` returns 0.5
  - (b) `acos(0.5)` returns  $1.0467 = (3.14)/3 = \pi/3$
- Tangent,  $y=\tan(x)$ , and Inverse Tangent,  $x=\text{atan}(y)$   
Examples:
  - (a) `tan(pi/4)` returns 1.0
  - (b) `atan(1.0)` returns  $0.7852 = (3.14)/4 = \pi/4$

### ❖ Application of Matrices

MATLAB functions operate on an array or matrix the same way as they operate on scalar values. These features are useful if we have a set of data to analyze.

- Example - Calculate the value of sine from  $-\pi$  to  $+\pi$  in  $\pi/10$  increments.

```
angle=[-pi : pi/10 : +pi];      % angle is an array containing -pi, -9pi/10, ..., 9pi/10, pi
sine_value = sin(angle);      % sine-value is an array where each element is the sine
                              % of corresponding element in angle array.
```

Executing the above commands results in `sine_value` containing 21 elements which are shown below:

```
sine_value = [-0.0000, -0.3090, -0.5878, -0.8090, -0.9511, -1.0000, -0.9511, -0.8090,
              -0.5878, -0.3090, 0, 0.3090, 0.5878, 0.8090, 0.9511, 1.0000, 0.9511, 0.8090,
              0.5878,
              0.3090, 0.0000]
```

### ❖ Graphing

MATLAB offers extensive graphics, including two- and three- dimensional plots as well as animated graphing in the form of movies. We will be focusing on the 2-dimensional capabilities, but students are encouraged to explore additional MATLAB capabilities shown in the graphics section demos.

The two foundational graphics functions are plot and stem.

➤ Plot

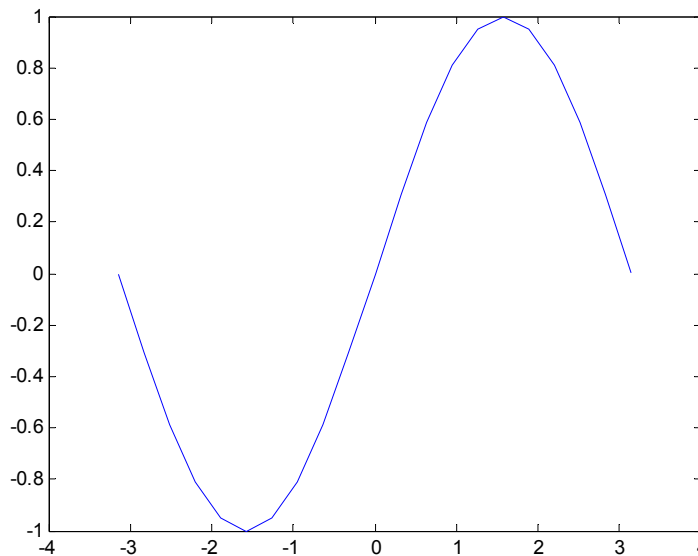
The plot function enables the designer to plot a two-axis line plot.

plot (x, y)        % plots vector y on vertical axis versus vector x in horizontal axis.  
                  % a one dimensional matrix (1xn or nx1) is referred to as vector.

- Example -- Calculate and plot the value of sine from  $-\pi$  to  $+\pi$  in  $\pi/10$  increments.

```
angle=[-pi : pi/10 : +pi];        % angle is an array containing  $-\pi, -9\pi/10, \dots, 9\pi/10, \pi$   
sine_value = sin(angle);        % sine-value is an array where each element is the sine  
                                  % of the corresponding element in the angle array.  
plot (angle, sine_value);        % function to plot sine_value versus angle.
```

The resulting plot is shown below:



➤ Annotating the Graph

When graphing data, it is important that the axes be clearly labeled and the graph title be specified. Additionally, it may be important to use a specific color and specific markers for each set of data. MATLAB allows the user to make all of these selections and more. For complete instructions on these and other graphing techniques refer to the help section of MATLAB.

As a means to demonstrate some of these features, let's do another example that adds more information to above graph:

- Example -- Calculate and plot the value of sine from  $-\pi$  to  $\pi$  in  $\pi/10$  increments. Label the axes and title the graph.

The following code is a full function that performs the calculation and plotting.

```
% full_plot.m  
% Author: Izad Khormaei  
% Last update: 9/10/06, Version: 1.1  
% Input parameters: n/a
```

```

% output parameter: n/a
% Description: Calculates and plots the value of sine from  $-\pi$  to  $+\pi$  in  $\pi/10$  increments.
% It will also label the axes and title of the plot.

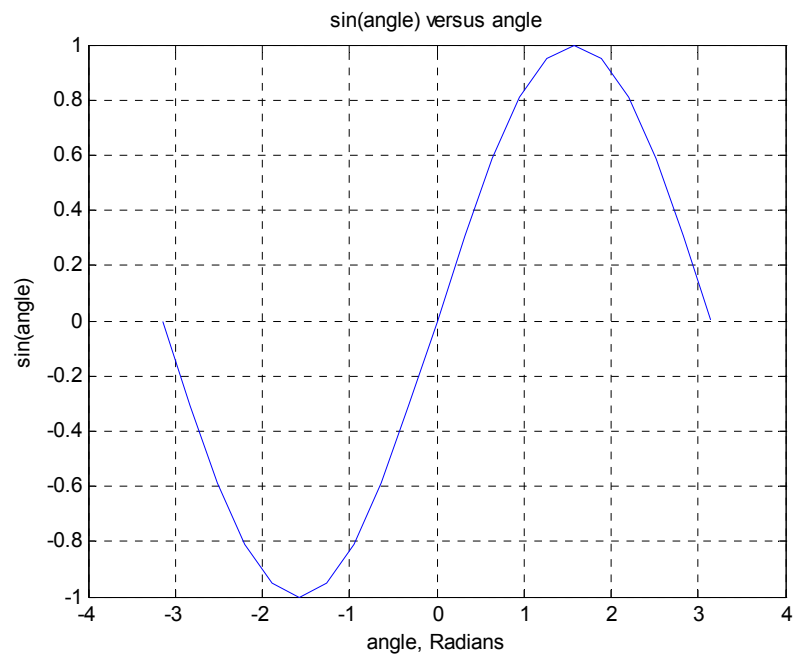
function [] = full_plot ()
    % Calculate the angle and sine
    angle=[-pi : pi/10 : +pi]; % angle is an array containing  $-\pi, -9\pi/10, \dots, 9\pi/10, \pi$ 
    sine_value = sin(angle); % sine-value is an array where each element is the sine
    % of the corresponding element in the angle array.

    plot (angle, sine_value); % plot sine_value versus the angle

    % Configure the plot
    title('sine(angle) versus angle') % Add title to the plot
    xlabel('angle, Radians') % Add horizontal axis label
    ylabel('sine(angle)') % Add vertical axis label
end % return full_plot()

```

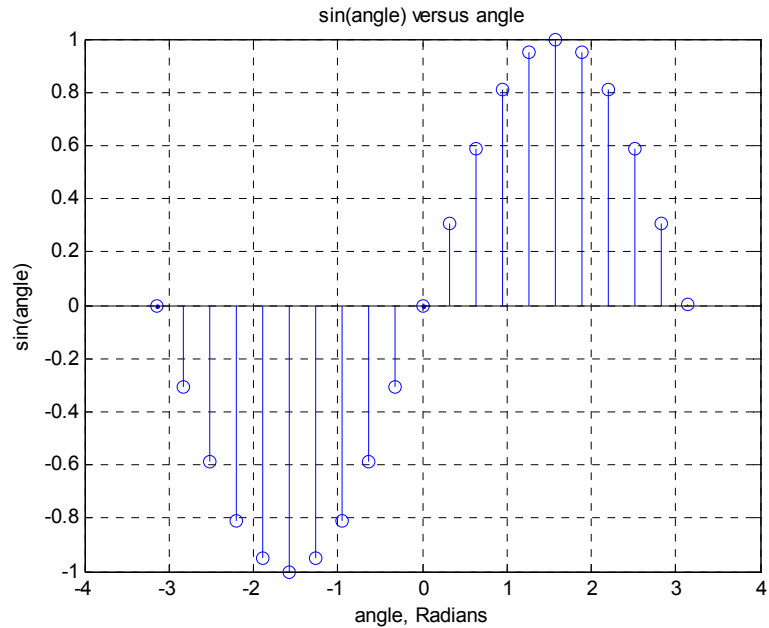
The resulting plot is shown below:



➤ **Stem () Function**

If the function plot() is replaced with stem() in previous , then we will see the results in a stem or discrete form which is shown below:





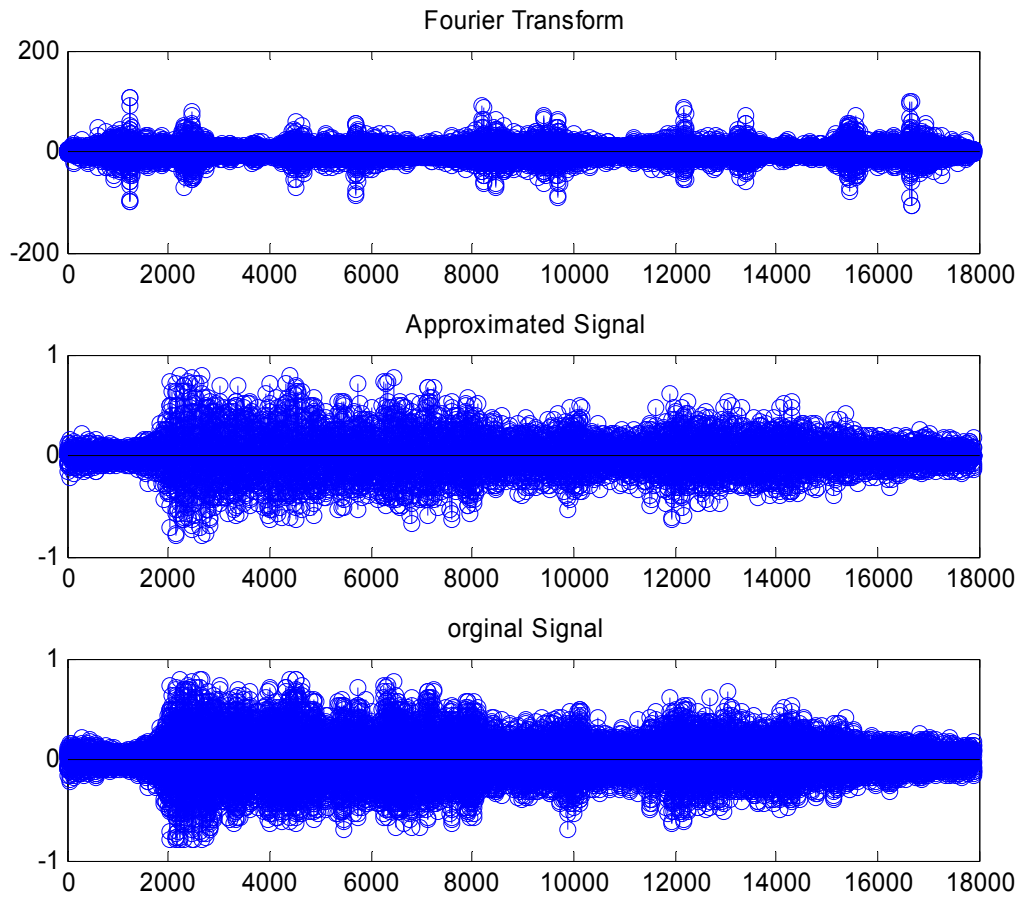
As mentioned earlier, there is an extensive level of graphics functionality available in MATLAB which would be valuable to explore in the future. The command “subplot” will be useful when attempting to plot multiple sets of data and compare them. The “subplot” function enables the user to plot multiple plots in a grid.

SUBPLOT (m,n,p), creates a grid of m-by-n matrix and places the next plot or stem graph in the p<sup>th</sup> cell of matrix. The cells are counted along the top row of the Figure window, then the first column of the second row, and so on.

For example, the following code generates a figure that contains three rows by one column of graphs:

```
subplot(3,1,1), stem(X), title(' Fourier Transform') % first plot row 1, col 1
subplot(3,1,2), stem(x), title(' Approximated Signal') % second plot row 2, col 1
subplot(3,1,3), stem(y(100:18000)), title(' original Signal'); % third plot row 3, col 1
```

The result of this code is show in the following figure.



## 7.6. Complex Numbers

Complex numbers came into existence in response to the fact that mathematicians needed to use  $\sqrt{-1}$  to solve quadratic equations. But in the real number paradigm,  $\sqrt{-1}$  is an invalid number, since there is no real number that when squared would equal  $-1$ .

In the 18<sup>th</sup> century, mathematicians came up with a solution to this dilemma by creating a new entity and referring to it as “i”. In engineering, we will use “j” to refer to this new entity since “i” is reserved for identifying current.

j is conveniently defined as:

$$j = \sqrt{-1} \quad \text{and} \quad j^2 = -1$$

### ❖ Complex Number General Form

The general form of a complex number has a real part and an imaginary part and is represented as:

$$C = x + jy \quad \text{Where:}$$

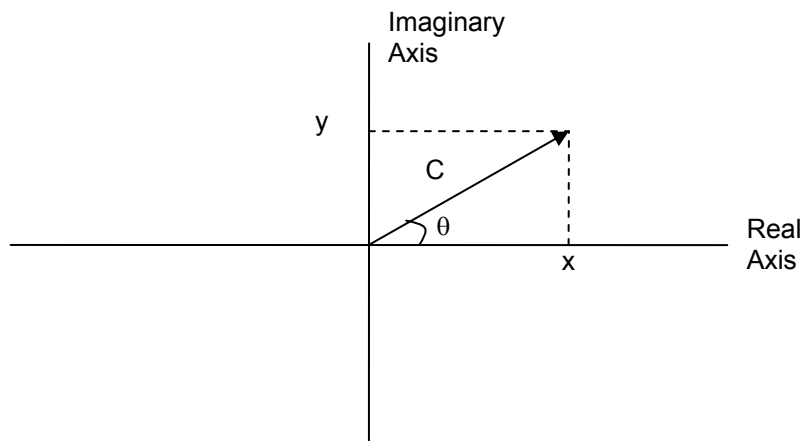
x is a real number constant and is referred to as the real part of C

y is a real number constant and is referred to as the imaginary part of C.

➤ For example  $C=2 - j6.2$  has a real part equal to 2 and an imaginary part equal to  $j6.2$ .

### ❖ Plotting Complex Numbers

Complex numbers are typically plotted on a plane where the vertical axis is the imaginary axis and the horizontal axis is real. The following diagram is a vector representing the complex number  $C=x + jy$ :



### ➤ Relationships

Here we can use trigonometry functions to describe relationships between various components such as the phase (angle,  $\theta$ ), real part, imaginary part and magnitude. For a complex number  $C=x + jy$ , the relationships are described below:

- The length of the complex number vector is called its magnitude. The complex number magnitude is defined by the following equation which uses the right triangle properties:  
magnitude of C =  $|C| = \sqrt{x^2 + y^2}$
- The phase or angle of a complex number is the angle “ $\theta$ ” between the real axis and the complex number vector. Utilizing the trigonometry functions, “ $\theta$ ” is defined by:

$$\theta = \arctan \frac{\text{Imaginary Part}}{\text{Real Part}} = \arctan \frac{y}{x} \quad \text{or} \quad \theta = \arcsin \frac{y}{|C|}$$

❖ Complex Number Arithmetic

➤ Addition and Subtraction

All the rules of real number arithmetic apply here, but it is mandatory to keep the imaginary and real parts separated. Add/subtract all the real numbers (real parts) and then add/subtract all of the coefficients of j (Imaginary parts). Below is an example:

$$(2 + j3.5) - (2.3 - j4.1) = (2 - 2.3) + j(3.5 + 4.1) = \underline{\underline{-0.3 + j7.6}}$$

➤ Multiplication

All of the rules of real number arithmetic apply here, and additionally, the definition  $j^2 = -1$  applies. Below is an example:

$$(2 + j4) * (3 - j5) = (2 * 3) - j(2 * 5) + j(4 * 3) - (4 * 5)(j^2) = 6 - j10 + j12 - (20)(-1) = \underline{\underline{26 + j2}}$$

- General form of multiplication can be shown as:  
 $(x + jy)(u + jv) = (xu - yv) + j(xv + yu)$

❖ Complex Number Application

Leonard Euler (pronounced Oiler), an influential 18<sup>th</sup> century Swiss mathematician and physicist, put forth the following relationship which has come to be known as Euler’s Identity:

$$e^{ja} = \cos a + j \sin a$$

This is a very important equation in engineering, since it is the general form of sinusoidal signals “cos(a)” as shown below:

$$\text{real part}\{e^{ja}\} = \cos a \quad \text{since } \sin(a) \text{ is the imaginary part}$$

In electrical engineering, devices such as capacitors and inductors change their behavior with signal frequency. Use of  $e^{ja}$  allows us to design and analyze circuits with time varying and frequency sensitive elements without using of calculus and differential equations.

There are two other ways of writing Euler’s Identity which may prove useful:

$$\cos a = \frac{e^{ja} + e^{-ja}}{2} \quad \& \quad \sin a = \frac{e^{ja} - e^{-ja}}{j2}$$

## 7.7. MATLAB Complex Number Operations

MATLAB operators support complex numbers, so all arithmetic operators can be used for complex numbers. MATLAB accepts both  $i$  and  $j$  as the representations of the imaginary ( $j=i=\sqrt{-1}$ ), but it defaults to  $i$  when displaying the results. For example:

$$C=(1 + j2)*(1 - j3) = 7 - j$$

MATLAB also provides the function “exp(x)” which evaluates  $e^x$ . Moreover, the following function proves useful when working with complex numbers:

- **imag(C) Function**  
This function returns the imaginary portion of a complex number “C”. For example:  
>> imag(10 + j12)  
returns the value 12, which is the imaginary part.
- **real (C) Function**  
This function returns the real portion of a complex number “C”. For example:  
>> real(10 + j12)  
returns the value 10, which is the real part.
- **abs(C) Function**  
This function returns the magnitude or absolute value of complex number “C”. For example:  
>> abs(10 + j12)  
returns the value 15.62 which is  $|10 + j12| = \sqrt{10^2 + 12^2} = 15.62$
- **angle(C) Function**  
This function returns the phase or angle of complex number “C”. For example:  
>> angle (10 + j12)  
return the value 0.88 which is the phase of the complex number =  $\tan^{-1}(12/10)$ .

## 7.8. MATLAB Exploration

Students are encouraged to explore the following functions:

- `sym("expression")`
- `int("expression", n)`
- `diff("expression", n)`
- `solve("equation1", "equation2", ...)`
- `mean(x)`
- `std(x)`

## 7.9. Additional Resources

- ❖ MathWorks. [MATLAB Reference Material Version R2000a](#). (2007) MathWorks

## 7.10. Problems

Instructions:

- Matlab or SciLab may be used for program development
- Homework solution should include flow chart and source code
- Be prepared to present your programs to the class

---

1. For the Following two Matrices

$$A = \begin{pmatrix} 2 & 3 & 4 & 5 \\ 1 & 0 & 0 & 3 \\ 7 & 8 & 4 & 9 \\ 3 & 2 & 8 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 5 & 4 & 2 \\ 7 & 4 & 5 & 3 \\ 0 & 1 & 4 & 4 \\ 3 & 3 & 0 & 1 \end{pmatrix}, \quad X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}, \quad D = \begin{pmatrix} 5 \\ -3 \\ -5 \\ 1 \end{pmatrix},$$

- a) Calculate  $C_1=A+B$ ,  $C_2=A-B$ ,  $C_3=A.*B$ ,  $C_4=A./B$  manually. Show your work.  
b) Use MATLAB to calculate  $C_1=A*B$  and  $C_2= A/B$ . include a copy of your code and results.  
c) Use MATLAB to find the value of X when  $D=2*B*X$

---

2. Calculate the answer to the following expression:

$$\begin{pmatrix} 5 \\ 12 \\ 9 \end{pmatrix} + \begin{pmatrix} 2 & -1 & 3 \\ 1 & 4 & 5 \\ 0 & 3 & 1 \end{pmatrix} * \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} =$$

---

3. Write the following matrix expression as system of equations:

$$\begin{pmatrix} 6 & -5 & -2 \\ 2 & 4 & 0 \\ 0 & 3 & -1 \end{pmatrix} * \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \\ 7 \end{pmatrix}$$

---

4. Calculate the answer for the following expression:

$$\begin{pmatrix} 5 \\ 12 \\ 9 \end{pmatrix} + \begin{pmatrix} 2 & -1 & 3 \\ 1 & 4 & 5 \\ 2 & 3 & 0 \end{pmatrix} * \begin{pmatrix} 1 \\ 4 \\ 0 \end{pmatrix} =$$

---

5. Write the following system of equations in a matrix form:

$$\begin{cases} 6x - 5y - 2z = 4 \\ 2x + 4y = 2 \\ 3y - z = 7 \end{cases}$$

---

6. Use MATLAB to plot  $f(t) = \sin(1000t) + \cos(250t)$  and use the “sound()” function to listen to it.



---

7. Use MATLAB's `sin()` and `sound()` function to test your hearing range. A human's range of hearing is between 20Hz - 20kHz. Generate the `sin()` values using 10 samples per period.

---

8. Given the complex number  $(10 + 15j) + 1/(12 - 10j)$ , write a program to calculate its magnitude, real part and imaginary part. Verify the results by manually calculating its magnitude, real part and imaginary part. Show your work.

*Hint: `abs()`, `angle()`, `real()` and `imag()` MATLAB functions may be useful.*

---

9. Given the complex number  $(10t + j6) / (12 - j3)$ , plot its magnitude and phase when  $t = [-10 : 0.1 : 10]$ . Title and label your plots.

---

10. Write a MATLAB function that accepts as input two angles in radians and displays the larger sine value followed by the smaller sine value.

```
function [ ] = sort_sines (x1, x2)
```

---

11. Write a MATLAB function that accepts as input three angles in radians, and displays the cosine values from smallest to the largest, using the following function framework:

```
function [ ] = sort_cosines (x1, x2, x3)
```

```
end
```

Hint: `cos(x)` function returns the cosine of angle  $x$ .

---

12. Write a program that accepts ten real numbers as input and displays the sorted number set from smallest to largest in scientific format.

*Hint: In this program, you are being asked to develop a sorting algorithm, which is a well researched area of computer science. You are encouraged to research existing knowledge in this area.*

---

13. Write a program that stores the 50 consecutive prime numbers starting with 2 in an array and displaying the prime numbers in a comma separated format. Use MATLAB to calculate the CPU time required to find the 50 consecutive prime numbers excluding display time. Explore algorithms to reduce the execution time.

## Appendix A. MATLAB and SCILAB Differences

The following table contain the most common differences encountered by students using SCILAB and MATLAB:

Matlab	SciLab Equivalent
From File menu open and create new source file. Source file has extension “.m”	From File menu open and create new source file. Source file has extension “.sci”
% when function returns a value function [r]= test(a, b) end	// when function returns a value function r= test(a, b) endfunction
% when function does not returns a value function []= test(a, b) end	// when function does not returns a value function test(a, b) endfunction
% comment	// comment
Write the function name in command window to run	Execute Menu and load into scilab and then Write the function name in command window to run.
mod(x,y)	pmodulo (x,y)
fprintf (1,'format string', var1, var2,...)	printf ('format string', var1, var2,...)

## Appendix B. Additional Resources

- ❖ Additional resources are available at the author's website <http://www.EngrCS.com/>