Prince Edward Island
CANADA

Education and Early Childhood Development
English Programs

Prince Edward Island Technology Education Curriculum

CURRICULUM

Career and Technical Education

Robotics 801A

2012
Prince Edward Island
Department of Education and
Early Childhood Development
Holman Centre
250 Water Street, Suite 101
Summerside, Prince Edward Island
Canada, C1N 1B6
Tel: (902) 438-4130
Fax: (902) 438-4062
www.gov.pe.ca/eecd/

Acknowledgments

The P.E.I. Department of Education and Early Childhood Development expresses its gratitude to members of the provincial curriculum committee for their professional expertise and insights in developing this Robotics 801A curriculum guide. In addition, the curriculum committee and pilot teachers who contributed comments and suggestions are to be commended for their commitment to developing exemplary science programs.

The Prince Edward Island Robotics Curriculum Committee

Kinkora Regional High School	Ian Gillis
Montague Senior High School	Jonathan Hayes
Morell Regional High School	Steven Wynne
Three Oaks Senior High	Jim Campbell
Westisle Composite High School	George Charchuk
	Randy Harper
Holland College	Debbie Hogan
	Keith Arsenault
Department of Education and Early Childhood Development	Ryan McAleer

Contents

Introduction

Background

The Prince Edward Island Department of Education and Early Childhood Development has developed a Career and Technical Education (CTE) Skilled Trades Pathway. This curriculum provides opportunities for students to achieve skills and knowledge towards occupations related to applied science and technolgy, engineering, and the skilled trades.

Robotics 801A is a career and technical education course that develops student scientific and technological knowledge and skills. It contains a balance of theory, design, and hands-on/minds-on activities that build students' scientific and technological literacy and help them to understand and appreciate scientific and technological concepts and processes.

This guide is intended to provide teachers with an overview of the outcomes framework for Robotics 801A. It also includes some suggestions to assist teachers in designing learning experiences and assessment tasks.

Focus and Context

Robotics design, programming, and construction is the primary context within which students will obtain knowledge and skills associated with Robotics 801A. In this course students will have an opportunity to engage in an engineering design process which will culminate in open-ended autonomous robot challenges. The goal of Robotics 801A is to promote technological competence, communication, and problem solving skills. Students will be encouraged to take pride and ownership in their learning journey through the development of a Personal Skills Logbook.

Aim

The aim of science education and career and technical education in Prince Edward Island is to foster the development of all learners as technologically literate and capable citizens who can develop, implement, and communicate practical, innovative, and responsible technological solutions to problems. This course is designed around the attitudes, skills, and knowledge students need to develop inquiry, problem-solving, and decision-making abilities and to understand the interrelationships among science, technology, society, and the environment (STSE).

Whether the student is acquiring new knowledge, increasing technological competence, assembling experimental apparatus, or constructing a new device, collaboration plays a very significant role in science and CTE. Consequently, teamwork is strategically incorporated in the Robotics 801A course design. The planning, programming, and building phases of robot design and construction explicitly involve students working in groups. The STSE project and the final course project (autonomous robot challenges) both culminate in group presentations, and peers can debate the merits of the presentation in a collaborative arena.

It is recommended that students have the opportunity to work through a series of team-building exercises at the beginning of this course. The purpose of these exercises is to have students recognize the importance of knowing the strengths of members of a team, communicating in the team dynamic, and viewing successes and failures as equally important in knowledge construction.

Program Design and Components

The Senior High School Learning Environment

The learning environment for grades 10-12 should be
• participatory, interactive, and collaborative;
• inclusive, caring, safe;
• challenging, inquiry based, issues oriented;
• a place where resource-based learning includes and encourages the multiple uses of technology, the media, and other visual texts as pathways to learning and as avenues for representing knowledge.

Learning environments are places where teachers
• integrate new ways of teaching and learning;
• have a variety of teaching and assessment strategies;
• value the place of dialogue in the learning process;
• recognize students as being intelligent in a number of different ways and encourage them to explore various ways of knowing by examining their strengths and working on their weaknesses;
• value the inclusive classroom and engage all learners in meaningful activities;
• acknowledge the ways in which gender, race, ethnicity, and culture shape particular ways of viewing and knowing the world;
• structure repeated opportunities for reflection so that reflection becomes an integral part of the learning process.

Learning should be extended to community facilities, allowing field trips and guest speakers to expand the learning environment.

Safety

Students and teachers need to feel safe, both physically and emotionally, in the school setting. In a learning environment where co-operative, active, and collaborative teaching strategies are utilized, students must become knowledgeable of their role in enabling a safe environment to exist.

Being empowered to take ownership for their own safety and that of their peers is an essential component of the classroom learning. Teachers can provide students with the knowledge required to prevent unnecessary risks in their learning environment. By being educated about the risk factors involved in the classroom setting, students can become active participants in the ownership of their own safety. In all learning situations, the teacher needs to encourage a positive, responsible student attitude toward safety.

While physical safety is of utmost importance in the classroom, emotional safety is equally important. Students need to know what constitutes acceptable and unacceptable behavior, and should be encouraged to be active learners without being intimidated by others.

Risk is involved in everything a person does. To minimize risk students must become conscious participants in ensuring a healthy, safe learning environment and must avoid complacent attitudes with regards to safety.

Learning and Teaching Science, Career and Technical Education

What students learn is fundamentally connected to how they learn it. The aim of scientific and technological literacy for all has created a need for new forms of classroom organization, communication, and instructional strategies. The teacher is a facilitator of learning whose major tasks include:

- creating a classroom environment to support the learning and teaching of science and CTE;
- designing effective learning experiences that help students achieve designated outcomes;
- stimulating and managing classroom discourse in support of student learning;
- learning about and then using students' motivations, interests, abilities, and learning styles to improve learning and teaching;
- assessing student learning, the scientific/technological tasks and activities involved, and the learning environment to make ongoing instructional decisions;
- selecting teaching strategies from a wide repertoire.

Effective science, career and technical education learning and teaching take place in a variety of situations. Instructional settings and strategies should create an environment that reflects a constructive, active view of the learning process. Learning occurs through actively constructing one's own meaning and assimilating new information to develop a new understanding.

The development of scientific and technological literacy in students is a function of the kinds of tasks they engage in, the discourse in which they participate, and the settings in which these activities occur. Students' disposition towards science and CTE is also shaped by these factors. Consequently, the aim of developing scientific and technological literacy requires careful attention to all of these facets of curriculum.

Learning experiences in science education and career and technical education should vary and should include opportunities for group and individual work, discussion among students as well as between teacher and students, and hands-on/minds-on activities that allow students to construct and evaluate explanations for the phenomena under investigation. Such investigations and the evaluation of the evidence accumulated provide opportunities for students to develop their understanding of the nature of science and the nature and status of CTE knowledge.

Methodology

Exemplary teachers use a variety of instructional strategies to meet the needs of all learners, and have the flexibility to call upon several different strategies both within one period and during a unit of study. Adolescent learners need a balance between practical work, listening, discussing, and problem solving. The teacher structures the learning situation and organizes necessary resources. In assessing the nature of the task, the teacher may find that the situation calls for teacher-directed activities with the whole class, small groups of students, or individual students.

As students develop a focus for their learning, the teacher moves to the perimeter to monitor learning experiences and to encourage flexibility and risk taking in the ways students approach learning tasks. The teacher intervenes, when appropriate, to provide support. In such an environment, students will feel central to the learning process.

As students accept more and more responsibility for learning, the teacher's role changes. The teacher notes what the students are learning and what they need to learn, and helps them to accomplish their tasks. The teacher can be a coach, a facilitator, a resource person, and a fellow learner. The teacher is a model whom students can emulate, and he/she instructs the student as needed during the learning process.

Through the whole process, the teacher is also an evaluator, assessing students' growth while helping them to recognize their achievements and their future needs.

In the CTE environment, the teacher acts as a facilitator to guide learning and assist students in their development of skills and abilities required to achieve the outcomes.

The workshop atmosphere provides freedom of movement and personal choice for the student. Accordingly, the student must develop a positive attitude and a level of responsibility equal to the challenge. An ideal working environment is one in which teachers and students enjoy mutual respect and trust.

Students often point to experiential activities as the best part of a program as they have the chance to work co-operatively and be actively involved in the learning process.

Organization/ Management of the CTE Workshop

In the organization and management of the physical space, the teacher must put some thought and planning into creating an environment that will be safe, inviting, stimulating, and interactive. The physical space needs to be clean, neat, and organized, so that students will maintain and respect the workshop area. A well-organized workshop environment will help the students develop respect for the facility, the teacher, and each other.

Indicators of an organized and safe learning environment include the following:

- Developing a Safety Code of Conduct
- Developing a cleanup schedule and procedure to ensure that the classroom is kept in good order and for students to develop a sense of responsibility for the maintenance and operation of the workshop
- Having a place for everything, and having everything in its place
- Maintaining tools in proper working condition, and ensuring all guards are in place
- Organizing tool cribs to name and display tool locations for easy accountability
- Maintaining power tools and equipment in top working condition
- Allowing adequate working space around all machines, and indicating safe zones on floor
- Organizing areas for both individual and small group collaboration
- Organizing space to maximize the vision or sight lines
- Maintaining an accurate inventory of tools, equipment and text resources required to support the curriculum
- Creating a list of suppliers and local businesses to support the consumable and curricular needs of the CTE program

Safety in Career and Technical Education

Safety in the CTE classroom is the number one priority of all teachers and students.

The Workers' Compensation Board of PEI states that "All accidents are preventable and avoidable."

CTE teachers must ensure the following:
- All fire regulations, exit procedures, and location of fire equipment are reviewed, posted, and clearly identified
- First-Aid stations, kits, and procedures are reviewed and clearly identified within the facility
- WHMIS sheets and information are reviewed and clearly identified
- All Occupational Health and Safety Regulations that apply to the particular CTE curriculum are followed
- All emergency shut-down and electrical override switches are clearly identified and accessible
- Scheduled maintenance on shop tools and equipment is completed
- A system for reporting safety concerns or issues is explained to the students
- A CTE Safety Code of Conduct is developed with and explained to each class

A CTE Safety Code of Conduct should be kept short and ideally be developed by each class as students tend to best adhere to codes of conduct they have had a hand in creating.

A CTE Safety Code of Conduct should be written in a positive manner to indicate the desired behavior (what students will do) as opposed to the undesired behavior (what students won't do).

A sample CTE Safety Code of Conduct appears below:

- We will respect others' right to work and learn in a safe environment
- We will report any safety concerns to the teacher
- We will wear safety glasses at all times when in the workshop
- We will wear appropriate clothing and required PPE
- We will maintain a neat and well-organized workspace
- We will ask for permission and/or instruction before using any tools equipment, or materials in the shop
- We will use common sense

Communicating in Science and CTE

Learning experiences should provide opportunities for students to use writing and other forms of representation as ways to learning. Students at all grade levels should be encouraged to use writing to speculate, theorize, summarize, discover connections, describe processes, express understandings, raise questions, and make sense of new information by using their own language as a step to the language of science and technology. Skills Logbooks are useful for such expressive and reflective writing. Purposeful note making is an intrinsic part of learning, helping students better record, organize, and understand information from a variety of sources. The process of creating webs, maps, charts, tables, graphs, drawings, and diagrams to represent data and results helps students learn, and also provides them with useful study tools.

Employing cross-curricular reading and writing strategies in the delivery of the curriculum will provide students with tools that will help them build knowledge and develop strategies to become more proficient in skills related to literacy, document use, reading text, and writing.

Learning experiences in science and CTE should also provide abundant opportunities for students to communicate their findings and understandings to others, both formally and informally, using a variety of forms for a range of purposes and audiences. Such experiences should encourage students to use effective ways of recording and conveying information and ideas and to use appropriate vocabulary in expressing their understandings. It is through opportunities to talk and write about the concepts they need to learn that students come to better understand both the concepts and the related vocabulary. It will be equally important for students to have exemplars of the strategies they need to develop and apply in selecting, constructing, and using various forms of communication.

The Three Processes of Scientific and Technological Literacy

An individual can be considered scientifically and technologically literate when he/she is familiar with, and able to engage in, three processes: inquiry, problem solving, and decision making.

Inquiry

Inquiry involves posing questions and developing explanations for phenomena. While there is general agreement that there is no such thing as a single scientific method or design model, students require certain skills to participate in the activities of science and CTE. Skills such as questioning, observing, inferring, predicting, measuring, hypothesizing, classifying, designing experiments, collecting data, analysing data, and interpreting data are fundamental to engaging in science and technology.

Problem Solving

The process of problem solving involves seeking solutions to human problems. It consists of proposing, creating, and testing prototypes, products, and techniques to determine the best solution to a given problem.

Decision Making

The process of decision making involves determining what we, as citizens, should do in a particular context or in response to a given situation. Decision making situations are important in their own right, and they also provide a relevant context for engaging in inquiry and/or problem solving.

Meeting the Needs of All Learners

Atlantic Canada Technology EducationFoundation Document and the ***Foundation for the Atlantic Canada Science Curriculum*** stress the need to design and implement a curriculum that provides equitable opportunities for all students according to their abilities, needs, and interests. Teachers must be aware of, and make adaptations to accommodate, the diverse range of learners in their classes. To adapt instructional strategies, assessment practices, and learning resources to the needs of all learners, teachers must create opportunities that will permit them to address their various learning styles.

As well, teachers must not only remain aware of and avoid gender and cultural biases in their teaching; they must also actively address cultural and gender stereotyping (e.g., about who is interested in and who can succeed in science and technology). Research supports the position that when curriculum is made personally meaningful and socially and culturally relevant, it is more engaging for groups traditionally under-represented, and indeed, for all students.

While this curriculum guide presents specific outcomes for each unit, it must be acknowledged that students will progress at different rates.

Teachers should provide materials and strategies that accommodate student diversity, and should validate students when they achieve the outcomes to the best of their abilities.

It is important that teachers articulate high expectations for all students and ensure that all students have equitable opportunities to experience success as they work toward achieving designated outcomes. Teachers should adapt classroom organization, teaching strategies, assessment practices, time, and learning resources to address students' needs and build on their strengths. The variety of learning experiences described in this guide provide access for a wide range of learners. Similarly, the suggestions for a variety of assessment practices provide multiple ways for learners to demonstrate their achievements.

Supporting EAL Learners

The Career and Technical Education curriculum is committed to the principle that learners of English as an additional language (EAL) should be full participants in all aspects of CTE. English deficiencies and cultural differences must not be barriers to full participation. All students should study a comprehensive curriculum with high-quality instruction and co-ordinated assessment.

All students, and EAL learners in particular, need to have opportunities and be given encouragement and support for speaking, writing, reading, and listening in CTE classes. Such efforts have the potential to help EAL learners overcome barriers, learn, and communicate effectively within the CTE curriculum.

To this end,

- schools should provide EAL learners with support in their dominant language and English language within the CTE classroom;
- teachers, counsellors, and other professionals should consider the English-language proficiency level of EAL learners as well as their prior course work in related CTE areas;
- the proficiency level of EAL learners should be solely based on their prior academic record and not on other factors;
- CTE teaching, curriculum, and assessment strategies should be based on best practices and build on the prior knowledge and experiences of students and on their cultural heritage;
- the nature of the CTE program should be communicated with appropriate language support to both students and parents;
- to verify that barriers have been removed, educators should monitor enrolment and achievement data to determine whether EAL learners have gained access to, and are succeeding in, CTE courses.

Education for Sustainable Development

Education for sustainable development (ESD) involves incorporating the key themes of sustainable development - such as poverty alleviation, human rights, health, environmental protection, and climate change - into the education system. ESD is a complex and evolving concept and requires learning about these key themes from a social, cultural, environmental, and economic perspective, and exploring how those factors are interrelated and interdependent.

With this in mind, it is important that all teachers, including CTE teachers, attempt to incorporate these key themes in their subject areas. One tool that can be used is the searchable on-line database ***Resources for Rethinking***, found at ***http://r4r.ca/en***. It provides teachers with access to materials that integrate ecological, social, and economic spheres through active, relevant, interdisciplinary learning.

Assessment and Evaluation

Assessment and evaluation require thoughtful planning and implementation to support the learning process and to inform teaching. All assessment and evaluation of student achievement must be based on the specific curriculum outcomes in the provincial curriculum.

Assessment involves the systematic collection of information about student learning with respect to:

- achievement of provincial curricula outcomes
- effectiveness of teaching strategies employed
- student self-reflection of learning.

Teachers are encouraged to be flexible in assessing the learning success of all students. Assessment criteria and the methods of demonstrating learning successes may vary from student to student depending on their strengths, interests, and learning styles.

Evaluation involves the weighting of the assessment information against a standard in order to make an evaluation or judgment about student achievement. Reporting of student achievement must be based on the achievement of curriculum outcomes.

There are three interrelated purposes of assessment. Each type of assessment, systematically implemented, contributes to an overall picture of an individual student's achievement.

Assessment for learning involves the use of information about student progress to support and improve student learning, inform instructional practices, and:

- is teacher-driven for student, teacher, and parent use
- occurs throughout the teaching and learning process, using a variety of tools
- engages teachers in providing differentiated instruction, feedback to students to enhance their learning, and information to parents in support of learning.

Assessment as learning actively involves student reflection on learning, monitoring of her/his own progress, and:

- supports students in critically analyzing learning related to curricular outcomes
- is student-driven with teacher guidance
- occurs throughout the learning process.

Assessment of learning involves teachers' use of evidence of student learning to make judgements about student achievement, and:

- provides opportunity to report evidence of achievement related to curricular outcomes
- occurs at the end of a learning cycle using a variety of tools
- provides the foundation for dicussions on placement or promotion

Assessment should reflect the full range of student learning in Robotics 801A; involve the use of a variety of information gathering strategies that allow teachers to address students' diverse backgrounds, learning styles, and needs; and provide students a variety of opportunities to demonstrate their learning.

Effective asessment strategies

- are explicit and are communicated to students and parent at the beginning of the school term (and at other appropriate points throughout the school year) so that students know expectations and criteria to be used to determine the level of achievement;
- must be valid in that they measure what they intend to measure;
- must be reliable in that they consistently achieve the same results when used again, or similar results with a similar group of students;
- involve students in the co-construction, interpretation, and reporting of assessment by incorporating their interests (students can select texts or investigate issues of personal interest);
- reflect where the students are in terms of learning a process or strategy, and help to determine what kind of support or instruction will follow;
- allow for relevant, descriptive, and supportive feedback that gives students clear directions for improvement;
- engage students in metacognitive self-assesment and goal setting that can increase their success as learners;
- are fair in terms of the students with the opportunity to demonstrate the extent and depth of their learning;
- accommodate the diverse needs of students with exceptionalities, including students with individual learning plans;
- assist teachers in selecting appropriate instruction and intervention strategies to promote the gradual release of responsibility;
- are transparent, pre-planned, and integrated with instruction as a component of the curriculum;
- are appropriate for the learning activities used, the purposes of instruction, and the needs and experiences of the students;

- are appropriate for the learning activities used, the purposes of instruction, and the needs and experiences of the students;
- are comprehensive and enable all students to have diverse and multiple opportunties to demonstrate their learning consistently, independently, and in a range of contexts in everyday instruction;
- include samples of students' work that provide evidence of their achievement;
- are varied in nature, administered over a period of time, and designed to provide opportunities for students to demonstrate the full range of their learning.

Reporting

Reporting on student learning should focus on the extent to which students have achieved the curriculum outcomes. Reporting involves communicating the summary and interpretation of information about student learning to various audiences who require it. Teachers have a special responsibility to explain accurately what progress students have made in their learning and to respond to parent and student inquiries about learning. Narrative reports might, for example, suggest ways in which students can improve their learning and identify ways in which teachers and parents can best provide support. Effective communication with parents regarding their children's progress is essential in fostering successful home-school partnerships. The report card in one means of reporting individual student progress. Other means include the use of conferences, notes, phone calls and electronic methods.

Assessment Techniques

Assessment techniques should match the style of learning and instruction employed. Several options are suggested in this curriculum guide from which teachers may choose, depending on the curriculum outcomes, class, and school/district policies. It is important that students know the purpose of an assessment, the method used, and the marking scheme being applied. In order that formative assessment support learning, the results, when reported to students, should indicate the improvements expected.

Assessment Techniques *Continued...*

Observation (formal or informal)

This technique provides a way of gathering information fairly quickly while a lesson is in progress. When used formally, the student(s) would be made aware of the observation and the criteria being assessed. Informally, it could be a frequent, but brief, check on a given criterion. Observation may offer information about the participation level of a student in a given task, use of a piece of equipment or software, or application of a given process. The results may be recorded in the form of checklists, rating scales, or brief written notes. It is important to plan in order that specific criteria are identified, suitable recording forms are ready, and that all students are observed within a reasonable period of time.

Performance

The Robotics 801A curriculum encourages learning through active participation. Many of the curriculum outcomes found in this guide promote skills and their application. There is a balance between process and content. In order that students appreciate the importance of skill development, it is critical that assessment provide feedback as they develop various skills. These may include the correct use of a piece of equipment or software; the application of an experimental technique; or the ability to interpret and follow instructions, or to research, organize, and present information. Assessing performance is most often achieved through observing the process.

Journal

Journals and logbooks provide an opportunity for students to express thoughts and ideas in a reflective way. By recording feelings, perceptions of success, and responses to new concepts, a student may be helped to identify his or her most effective learning style.

Knowing how to learn in an effective way is powerful information. Journal entries also give indicators of developing attitudes to CTE concepts, processes, and skills, and how these may be applied in the context of society. Self-assessment, through a journal or logbook, permits a student to consider strengths and weaknesses, attitudes, interests, and new ideas. Teachers can use the logbook to provide feedback on students' progress relative to the specific curriculum outcomes, tasks, and sub-tasks related to the course.

Interview

This curriculum promotes understanding and applying CTE concepts. Interviewing a student allows the teacher to confirm that learning has taken place beyond factual recall. Discussion allows a student to display an ability to use information and clarify understanding. Interviews may be brief discussions between teacher and student or they may be more extensive and include student, parent, and teacher. Such conferences allow a student to be pro-active in displaying understanding. It is helpful for students to know which criteria will be used to assess formal interviews. This assessment technique provides an opportunity to students whose verbal presentation skills are stronger than their writing skills.

Assessment Techniques *Continued...*

Paper and Pencil (assignment or test)

These techniques can be formative or summative. Several curriculum outcomes call for displaying ideas, data, conclusions, and the results of practical or literature research. These can be in written form for display or for direct teacher assessment. Whether an activity/product is part of learning or a final statement, students should know the expectations for the exercise and the rubric by which it will be assessed. Written assignments and tests can be used to assess knowledge, understanding, and application of concepts. They are less effective in assessing skills, processes, and attitudes. The purpose of the assessment should determine what form of paper and pencil exercise is used.

Presentation

This curriculum includes outcomes that require students to analyse and interpret information; identify relationships among science, technology, society, and environment; to be able to work in teams; and communicate information. Although the process can be time consuming, these activities are best displayed and assessed through presentations. These can be given orally, in written/pictorial form, by project summary, or by using electronic systems such as video or computer software. Whatever the level of complexity or format used, it is important to consider the curriculum outcomes as a guide to assessing the presentation. The outcomes indicate the process, concepts, and context for which and about which a presentation is made.

Portfolio

Portfolios offer another option for assessing student progress in meeting curriculum outcomes over a more extended period of time. The Skill Logbook is an integral part of any Career and Technical Education portfolio. The Skill Logbook is a means by which students can reflect on their learning and teachers can provide feedback on students' progress relative to specific curriculum outcomes and tasks related to the course. This form of assessment allows the student to be central in the process. The Skill Logbook is intended to be managed and built by the students.

Decisions about the portfolio and its contents can be made by the student. What is placed in the portfolio, the criteria for selection, how the portfolio is used, how and where it is stored, and how it is evaluated are some of the questions to consider when planning to collect and display student work in this way. The portfolio should provide a long-term record of growth in learning and skills. This record of growth is important for individual reflection and self-assessment, but it is also important to share with others. For many students it is exciting to review a portfolio and see the record of development over time. Students may enhance their portfolios previously created in grade 9, wish to use Career Cruising as an electronic means to store portfolio information.

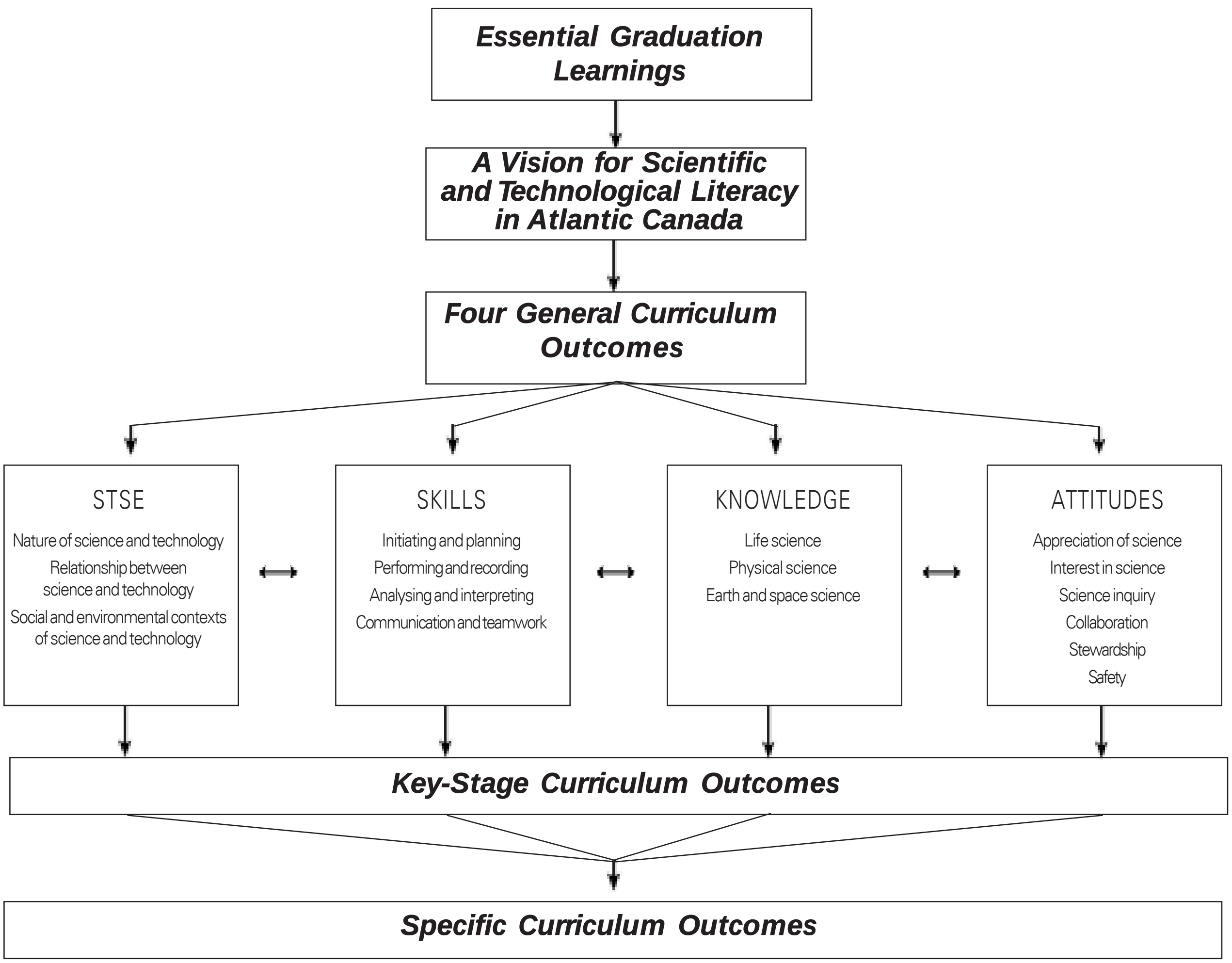
Curriculum Outcomes Framework

Overview

The Robotics 801A curriculum is based on an outcomes framework that includes statements of essential graduation learnings, general curriculum outcomes, key-stage curriculum outcomes, and specific curriculum outcomes. The general, key-stage, and specific curriculum outcomes reflect components of the pan-Canadian C***ommon Framework of Science Learning Outcomes K to 12*** and the ***Atlantic Canada Technology Education Foundation Document***. The diagram below provides the blueprint of the outcomes framework.

Outcomes Framework

FIGURE 1



Essential Graduation Learnings

Essential graduation learnings are statements describing the knowledge, skills, and attitudes expected of all students who graduate from high school. Achievement of the essential graduation learnings will prepare students to continue to learn throughout their lives. These learnings describe expectations not in terms of individual school subjects but in terms of knowledge, skills, and attitudes developed throughout the curriculum. They confirm that students need to make connections and develop abilities across subject boundaries, and be ready to meet the shifting and ongoing opportunities, responsibilities, and demands of life after graduation. The essential graduation learnings are the following:

Aesthetic Expression

Graduates will be able to respond with critical awareness to various forms of the arts and be able to express themselves through the arts.

Citizenship

Graduates will be able to assess social, cultural, economic, and environmental interdependence in a local and global context.

Communication

Graduates will be able to use the listening, viewing, speaking, reading, and writing modes of language(s), as well as mathematical and scientific concepts and symbols, to think, learn, and communicate effectively.

Personal Development

Graduates will be able to continue to learn and to pursue an active, healthy lifestyle.

Problem Solving

Graduates will be able to use the strategies and processes needed to solve a wide variety of problems, including those requiring linguistic, mathematical, and scientific concepts.

Technological Competence

Graduates will be able to use a variety of technologies, demonstrate an understanding of technological applications, and apply appropriate technologies for solving problems.

General Curriculum Outcomes

The general curriculum outcomes form the basis of the outcomes framework. They also identify the key components of scientific literacy. Four general curriculum outcomes have been identified to delineate the four critical aspects of students' scientific literacy. They reflect the wholeness and interconnectedness of learning and should be considered interrelated and mutually supportive.

Science, Technology, Society, and the Environment

Students will develop an understanding of the nature of science and technology, of the relationships between science and technology, and of the social and environmental contexts of science and technology.

Skills

Students will develop the skills required for scientific and technological inquiry, for solving problems, for communicating scientific ideas and results, for working collaboratively, and for making informed decisions.

Knowledge

Students will construct knowledge and understandings of concepts in life science, physical science, and Earth and space science, and apply these understandings to interpret, integrate, and extend their knowledge.

Attitudes

Students will be encouraged to develop attitudes that support the responsible acquisition and application of scientific and technological knowledge to the mutual benefit of self, society, and the environment.

Key-Stage Curriculum Outcomes

Key-stage curriculum outcomes are statements that identify what students are expected to know, be able to do, and value by the end of grades 3, 6, 9, and 12 as a result of their cumulative learning experiences in science. The key-stage curriculum outcomes are from the ***Common Framework for Science Learning Outcomes K to 12***.

Specific Curriculum Outcomes

Specific curriculum outcome statements describe what students are expected to know and be able to do at each grade level. They are intended to help teachers design learning experiences and assessment tasks. Specific curriculum outcomes represent a framework for assisting students to achieve the key-stage curriculum outcomes, the general curriculum outcomes, and ultimately, the essential graduation learnings. Specific curriculum outcomes are organized in units for each grade level.

Attitude Outcomes

It is expected that the Atlantic Canada science program will foster certain attitudes in students throughout their school years. The STSE, skills, and knowledge outcomes contribute to the development of attitudes, and opportunities for fostering these attitudes are highlighted in the Elaborations—Strategies for Learning and Teaching sections of each unit.

Attitudes refer to generalized aspects of behaviour that teachers model for students by example and by selective approval. Attitudes are not acquired in the same way as skills and knowledge. The development of positive attitudes plays an important role in students' growth by interacting with their intellectual development and by creating a readiness for responsible application of what students learn.

Since attitudes are not acquired in the same way as skills and knowledge, outcome statements for attitudes are written as key-stage curriculum outcomes for the end of grades 3, 6, 9, and 12. These outcome statements are meant to guide teachers in creating a learning environment that fosters positive attitudes.

The following pages present the attitude outcomes from the pan-Canadian ***Common Framework of Science Learning Outcomes K to 12*** for the end of grade 12.

Common Framework of Science Learning Outcomes K to 12
Attitude Outcome Statements

By the end of grade 12, it is expected that students will be encouraged to

Appreciation of Science	*Interest in Science*	*Scientific Inquiry*
436 value the role and contribution of science and technology in our understanding of phenomena that are directly observable and those that are not ***43***7 appreciate that the applications of science and technology can raise ethical dilemmas ***438*** value the contributions to scientific and technological development made by women and men from many societies and cultural backgrounds ***Evident when students, for example,*** • consider the social and cultural contexts in which a theory developed • use a multi-perspective approach, considering scientific, technological, economic, cultural, political, and environmental factors when formulating conclusions, solving problems, or making decisions on STSE issues • recognize the usefulness of being skilled in mathematics and problem solving • recognize how scientific problem solving and the development of new technologies are related • recognize the contribution of science and technology to the progress of civilizations • carefully research and openly discuss ethical dilemmas associated with the applications of science and technology • show support for the development of information technologies and science as they relate to human needs • recognize that western approaches to science are not the only ways of viewing the universe • consider the research of both men and women	***439*** show a continuing and more informed curiosity and interest in science and science-related issues ***440*** acquire, with interest and confidence, additional science knowledge and skills using a variety of resources and methods, including formal research ***441*** consider further studies and careers in science- and explore where further science- and technology-related fields ***Evident when students, for example,*** • conduct research to answer their own questions • recognize that part-time jobs require science- and technology-related knowledge and skills • maintain interest in or pursue further studies in science • recognize the importance of making connections between various science disciplines • explore and use a variety of methods and resources to increase their own knowledge and skills • are interested in science and technology topics not directly related to their formal studies • explore where further science- and technology-related studies can be pursued • are critical and constructive when considering new theories and techniques • use scientific vocabulary and principles in everyday discussions • readily investigate STSE issues	***442*** confidently evaluate evidence and consider alternative perspectives, ideas, and explanations ***443*** use factual information and rational explanations when analysing and evaluating ***444*** value the processes for drawing conclusions ***Evident when students, for example,*** • insist on evidence before accepting a new idea or explanation, and ask questions and conduct research to confirm and extend their understanding • criticize arguments based on the faulty, incomplete, or misleading use of numbers • recognize the importance of reviewing the basic assumptions from which a line of inquiry has arisen • expend the effort and time needed to make valid inferences • critically evaluate inferences and conclusions, cognizant of the many variables involved in experimentation • critically assess their opinion of the value of science and its applications • criticize arguments in which evidence, explanations, or positions do not reflect the diversity of perspectives that exist • insist that the critical assumptions behind any line of reasoning be made explicit so that the validity of the position taken can be judged • seek new models, explanations, and theories when confronted with discrepant events or evidence

Common Framework of Science Learning Outcomes K to 12

Attitude Outcome Statements (continued)

By the end of grade 12, it is expected that students will be encouraged to

Collaboration	Stewardship	Safety in Science
445 work collaboratively in planning and carrying out investigations, as well as in generating and evaluating ideas ***Evident when students, for example,*** • willingly work with any classmate or group of individuals regardless of their age, gender, or physical and cultural characteristics • assume a variety of roles within a group, as required • accept responsibility for any task that helps the group complete an activity • give the same attention and energy to the group's product as they would to a personal assignment • are attentive when others speak • are capable of suspending personal views when evaluating suggestions made by a group • seek the points of view of others and consider diverse perspectives • accept constructive criticism when sharing their ideas or points of view • criticize the ideas of their peers without criticizing the persons • evaluate the ideas of others objectively • encourage the use of procedures that enable everyone, regardless of gender or cultural background, to participate in decision making • contribute to peaceful conflict resolution • encourage the use of a variety of communication strategies during group work • share the responsibility for errors made or difficulties encountered by the group	***446*** have a sense of personal and shared responsibility for maintaining a sustainable environment ***447*** project the personal, social, and environmental consequences of proposed action ***448*** want to take action for maintaining a sustainable environment ***Evident when students, for example,*** • willingly evaluate the impact of their own choices or the choices scientists make when they carry out an investigation • assume part of the collective responsibility for the impact of humans on the environment • participate in civic activities related to the preservation and judicious use of the environment and its resources • encourage their peers or members of their community to participate in a project related to sustainability • consider all perspectives when addressing issues, weighing scientific, technological, and ecological factors • participate in social and political systems that influence environmental policy in their community • examine/recognize both the positive and negative effects on human beings and society of environmental changes caused by nature and by humans • willingly promote actions that are not injurious to the environment • make personal decisions based on a feeling of responsibility toward less privileged parts of the global community and toward future generations • are critical-minded regarding the short- and long-term consequences of sustainability	***449*** show concern for safety and accept the need for rules and regulations ***450*** be aware of the direct and indirect consequences of their actions ***Evident when students, for example,*** • read the labels on materials before using them, interpret the WHMIS symbols, and consult a reference document if safety symbols are not understood • criticize a procedure, a design, or materials that are not safe or that could have a negative impact on the environment • consider safety a positive limiting factor in scientific and technological endeavours • carefully manipulate materials, cognizant of the risks and potential consequences of their own actions • write into a laboratory procedure safety and waste-disposal concerns • evaluate the long-term impact of safety and waste disposal on the environment and the quality of life of living organisms • use safety and waste disposal as criteria for evaluating an experiment • assume responsibility for the safety of all those who share a common working environment by cleaning up after an activity and disposing of materials in a safe place • seek assistance immediately for any first-aid concerns such as cuts, burns, or unusual reactions • keep the work station uncluttered, with only appropriate lab materials present

Curriculum Guide Organization

Specific curriculum outcomes are organized in sections in this curriculum guide. Suggestions for learning, teaching, assessment, and resources are provided to support student achievement of the outcomes. Suggested times for each topic are also provided. Although Robotics 801A is 110 hours (~90 classes) in duration, the cumulative topic time allocated is 98 hours (~80 classes). The remaining 12 hours allow for summative assessment considerations and robot performance.

The order in which the sections appear in this guide is meant to suggest a sequence. In some cases, the rationale for the recommended sequence is related to the conceptual flow across the semester. Some units or certain aspects of units may also be combined or integrated. This is one way of assisting students as they attempt to make connections across topics in science and technology or between science, technology, and the real world.

The numbering system used is from the pan-Canadian document:

- 100s—Science-Technology-Society-Environment (STSE) outcomes
- 200s—Skills outcomes
- 300s—Knowledge outcomes
- 400s—Attitude outcomes (see pages 18-20)

These code numbers appear in brackets after each specific curriculum outcome (SCO).

Section Overviews	Suggested Timeframe	Number of Outcomes	Number of Outcomes by Cognitive Level		
			K*	U & A*	HMP*
Science, Technology, Society, and the Environment	5 Classes	7	2	2	3
Student Reflection, Skills Logbook	Ongoing	3	0	1	2
Introduction to Autonomous Robotics	3 Classes	4	3	1	0
Drivetrain, Microcontroller, and Transmitter	4 Classes	3	0	2	1
Compound Gears	7 Classes	3	0	2	1
Centre of Gravity and Degrees of Freedom	7 Classes	3	0	2	1
Introduction to Sensors	3 Classes	8	2	4	2
Introduction to Programming	6 Classes	8	3	4	1
Autonomous Guided Builds	15 Classes	10	0	10	0
Final Project	30 Classes	4	0	2	2
TOTALS	80 Classes	53	10	30	13

* K - Knowledge, U & A - Understanding & Application, HMP - Higher Mental Process

The Four-Column Spread

All units have a two-page layout of four columns as illustrated below. In some cases, the four-column spread continues to the next two-page layout. Outcomes are grouped by a topic indicated at the top of the left page.

Two-Page, Four-Column Spread

Page One		***Page Two***	
Topic			
Outcomes	Elaborations—Strategies for Learning and Teaching	Tasks for Instruction and/or Assessment	Resources/Notes
Students will be expected to • Specific curriculum outcome based on the pan-Canadian outcomes (outcome number) • Specific curriculum outcome based on the pan-Canadian outcomes (outcome number)	elaboration of outcome and strategies for learning and teaching elaboration of outcome and strategies for learning and teaching	***Informal/Formal Observation*** ***Performance*** ***Journal*** ***Interview*** ***Paper and Pencil*** ***Presentation*** ***Portfolio***	Useful teacher resources

Column One: Outcomes

The first column provides the specific curriculum outcomes. These are based, in part, on the pan-Canadian ***Common Framework of Science Learning Outcomes K to 12***. The statements involve the Science-Technology-Society-Environment (STSE), skills, and knowledge outcomes indicated by the outcome number(s) that appear(s) in parentheses after the outcome. It should be noted that select outcomes contain a list of directives to further delineate the outcome for instruction and assessment purposes. Select STSE and skills outcomes have been written in a context that shows how these outcomes should be addressed. Furthermore, the 300 level knowledge outcomes are specific to Prince Edward Island and are not reflected in the pan-Canadian ***Common Framework of Science Learning Outcomes K to 12.***

Specific curriculum outcomes have been grouped by topic. Other groupings of outcomes are possible and in some cases may be necessary in order to take advantage of local situations. The grouping of outcomes provides a suggested teaching sequence. Teachers may prefer to plan their own teaching sequence to meet the learning needs of their students.

Column one and column two define what students are expected to learn, and be able to do.

Column Two: Elaborations—Strategies for Learning and Teaching

The second column may include elaborations of outcomes listed in column one, and describes learning environments and experiences that will support students' learning.

The strategies in this column are intended to provide a holistic approach to instruction. In some cases, they address a single outcome; in other cases, they address a group of outcomes.

Column Three: Tasks for Instruction and/or Assessment

The third column provides suggestions for ways that students' achievement of the outcomes could be assessed. These suggestions reflect a variety of assessment techniques and materials that include, but are not limited to, informal/formal observation, performance, journal, interview, paper and pencil, presentation, and portfolio. Some assessment tasks may be used to assess student learning in relation to a single outcome, others to assess student learning in relation to several outcomes. The assessment item identifies the outcome(s) addressed by the outcome number in brackets after the item.

Column Four: Resources/Notes

This column provides an opportunity for teachers to make note of useful resources.

Section Overviews/Specific Curriculum Outcomes

Science, Technology, Society, and the Environment

- analyse why and how a particular robotics technology was developed and improved over time (115-5)
- describe the functioning of domestic and industrial autonomous robotic technologies that are an integral part of our lives and community (116-5/117-5)
- identify and describe science- and technology-based occupations related to this course (117-7)
- evaluate the design of a robotics technology and the way it functions on the basis of identified criteria, such as safety, cost, availability, and impact on everyday life and the environment (118-3)
- use library and electronic research tools to collect information on a given topic (213-6)
- communicate questions, ideas, and intentions, and receive, interpret, understand, support, and respond to the ideas of others (215-1)

Student Reflection, Skills Logbook

- evaluate individual and group processes used in planning, problem solving, decision making, and completing a task (215-7)
- analyse the knowledge and skills acquired in their study of science, to identify areas of further study related to science and technology (117-9)
- select and use appropriate numeric, symbolic, graphical, and linguistic modes of representation to communicate ideas, plans, and results (215-2)

Introduction to Autonomous Robotics

- provide examples of how science and technology are an integral part of their lives and their community (117-5)
- explain the similarities and differences between radio-controlled and autonomous robots (300-1)
- identify and describe the basic components of radio-controlled and autonomous robotics subsystems (300-2)
- analyse how robotics subsystems interact (300-3)

Drivetrain, Microcontroller, and Transmitter

- evaluate the design and function of a squarebot robot in terms of stability, propulsion, speed, and control (118-4a)
- configure a transmitter and microcontroller effectively and accurately to control a robot (213-3a)
- collaborate with team members to construct and test a squarebot robot, and troubleshoot problems as they arise. (214-14/215-6 a)

Compound Gears

- design and construct a device that manipulates the relationship between gearing, speed, and torque (212-3a)
- collaborate with team members to develop and implement a plan, and troubleshoot problems as they arise (215-6)
- qualitatively describe and quantitatively calculate the gear ratios of compound gear trains (300-5)

Centre of Gravity and Degrees of Freedom

- determine a robot's centre of gravity (212-3b)
- collaborate with team members to construct and test a robotic manipulator, and troubleshoot problems as they arise (214-14/215-6 b)
- analyse how the stability and dexterity of a robotic manipulator are affected by its centre of gravity and degrees of freedom (300-6)

Introduction to Sensors

- describe the functioning of domestic and industrial autonomous robotic technologies that are an integral part of our lives and community (116-5/117-5)
- construct arguments to support a decision or judgement, using examples and evidence and recognizing various perspectives (118-6)
- communicate questions, ideas, and intentions, and receive, interpret, understand, support, and respond to the ideas of others (215-1)
- identify multiple perspectives that influence a science-related decision or issue (215-4)
- explain how the sensor subsytem functions and how it interacts with other robot subsystems (300-7)
- define the difference between analog and digital sensors and specify applications for each (300-8)
- identify and describe the purpose and function of each VEX sensor (300-9)
- analyse a given scenario and determine what sensors could be used to accomplish the task (300-10)

Section Overviews/Specific Curriculum Outcomes

Introduction to Programming

- collaborate with team members to develop and implement a plan, and troubleshoot problems as they arise (215-6)
- evaluate individual and group processes used in planning, problem solving and decision making, and completing a task (215-7)
- explain programming (300-11)
- identify and describe programming syntax and keywords required to write a program (300-12)
- create flowcharts involving sub-programs to break a complicated behaviour into a series of simple tasks (300-13)
- manipulate and use program flow statements and functions to program a robot to complete a task (300-14)
- diagnose the different inputs and outputs that can be connected to the robot logic subsystem (300-15)
- program a robot to operate in radio controlled, autonomous, and combined setups (300-16)

Autonomous Guided Builds

- carry out procedures controlling the major variables and adapting or extending procedures where required (213-2)
- use contact sensors effectively and accurately for collecting data to assist in the control of robot, or robot component, movement (213-3b)
- use a light sensor effectively and accurately for collecting data to assist in the control of robot, or robot component, movement (213-3c)
- use an ultrasonic sensor effectively and accurately for collecting data to assist in the control of robot, or robot component, movement (213-3d)
- use a position sensor effectively and accurately for collecting data to assist in the control of robot, or robot component, movement (213-3e)
- collaborate with team members to construct and test a robot with autonomous features, and trobleshoot problems as they arise (214-14/215-6 c)
- identify new applications of a contact sensor that arise from what was learned (214-17a)
- identify new applications of an optical sensor that arise from what was learned (214-17b)
- identify new applications of an ultrasonic sensor that arise from what was learned (214-17c)
- identify new applications of a position sensor that arise from what was learned (214-17d)

Final Project

- design and construct a robot with autonomous and RC functions (212-3c)
- collaborate with team members to construct and test a robot design, using components and sensors conducive to completing a predetermined task, and troubleshooting problems as they arise (214-14/215-6 d)
- propose alternative solutions to a given practical problem, identify the potential strengths and weaknesses of each, and select one as the basis for a plan (214-15)
- select and use appropriate numeric, symbolic, graphical, and linguistic modes of representation to communicate ideas, plans, and results (215-2)

Robotics 801A
Curriculum Sections

Science, Technology, Society, and the Environment
~5 Classes

Outcomes

Students will be expected to

- ***analyse why and how a particular robotics technology was developed and improved over time (115-5)***
- ***describe the functioning of domestic and industrial autonomous robotic technologies that are an integral part of our lives and community (116-5/117-5)***
- ***identify and describe science- and technology-based occupations related to this course (117-7)***
- ***evaluate the design of a robotics technology and the way it functions on the basis of identified criteria, such as safety, cost, availability, and impact on everyday life and the environment (118-3)***
- ***use library and electronic research tools to collect information on a given topic (213-6)***
- ***communicate questions, ideas, and intentions, and receive, interpret, understand, support, and respond to the ideas of others (215-1)***

Elaborations—Strategies for Learning and Teaching

The outcomes addressed in the "Science, technology, Society, and the Environment" (STSE) section (pp. 28-29) are project-related and can be addressed at any time throughout this course. However, some components of the project (programming flowchart) will be better appreciated after they are addressed later in the course. The section entitled "Student Reflection, Skills Portfolio" (pp. 30-31) contains outcomes that are expected to be addressed continuously throughout this course. It is recommended that all remaining curriculum guide sections be addressed in order as they appear.

Specific curriculum outcomes 115-5, 116-5, 117-5, 117-7, 118-3, and 213-6 can be addressed by providing students with an opportunity to engage, individually or collaboratively, in a research project that would require them to use print and electronic resources to research, select, and integrate information on a robotics technology.

There are two suggested options for this project. In one option students could be asked to select an existing robotics technology of interest; describe its use, history, and evolution; create a flowchart that outlines the programming of the robot's functions; and consider from an array of perspectives (e.g., economy, society, and the environment) the advantages and disadvantages of the technology.

In another option students could be asked to identify a need for a RC or autonomous robot to help a particular group in society; describe its use and the demographic that it will serve; create a flowchart that outlines the programming of the robot's functions; and consider from an array of perspectives (e.g., economy, society, and the environment) the advantages and disadvantages of the technology with respect to what is presently available in society.

Students should be asked to present their STSE projects to the class. Through questioning, students will be exposed to various other perspectives and, as a result, will be expected to better appreciate the importance of communication and review in the development of robotics technology.

Science, Technology, Society, and the Environment

~5 Classes

Tasks for Instruction and/or Assessment

Performance

- Research an existing or proposed robotics technology of interest. Describe the technology, discuss its existing or proposed use, along with its advantages and disadvantages, as viewed from different perspectives (economy, society, and the environment). Provide a brief history of your robot or societal need. Use the instructions/questions in Appendix B to guide your research. (115-5, 116-5, 117-5, 117-7, 118-3, 213-6)

Presentation

- Prepare an electronic multimedia presentation to assist in communicating your information on a robotics technology of interest. (215-1)

- Respond to the project ideas presented by other students. Ask questions of the presenters that demonstrate your understanding or interpretation of their ideas. (215-1)

Paper and Pencil

- Identify the possible uses of a robotics technology presented in your class. Describe the two most pertinent advantages and disadvantages of the robot. (118-3, 215-1)

- Of the robotics technologies presented, which do you believe are most suitable to meet a societal need. Support your decision with the evidence that was presented or discussed. (118-3, 215-1)

Resources/Notes

Appendix B: STSE Research Project

STSE Project
Presentation Rubric
Report Form

Student Reflection, Skills Logbook

Outcomes

Students will be expected to

- ***evaluate individual and group processes used in planning, problem solving, decision making, and completing a task (215-7)***

- ***analyse the knowledge and skills acquired in their study of robotics technology, to identify areas of further study related to science and robotics technology (117-9)***

- ***select and use appropriate numeric, symbolic, graphical, and linguistic modes of representation to communicate ideas, plans, and results (215-2)***

Elaborations—Strategies for Learning and Teaching

The outcomes addressed in the "Student Reflection, Skills Logbook" (pp. 30-31) section are expected to be addressed continuously throughout this course.
Most of the activities in Robotics 801A involve team collaboration during which student teams engage in planning, problem solving, decision making, and completing a task. Students should be asked periodically (e.g., quarterly) to explicitly evaluate their participation and contribution to this process. Similarly, students should be provided with the opportunity to assess their peers. It is recommended that self- and peer-assessment rubrics be provided to students at the beginning of the course. Furthermore, it is recommended that students be involved in the construction of these rubrics so that they will have ownership and a better awareness of the expectations. Teachers should conduct formal interviews with each student to reflect upon self- and peer-assessments, discuss strengths, and identify strategies for personal growth.

Students will be required to maintain a skills logbook of acquired and desired skills. Teachers should spend some time explaining the technical skills rubric scale and logbook. Teachers should model appropraite examples of evidence of learning and learning journal entries. The logbook could be collected several times (quarterly reporting periods) throughout the duration of this course. It is expected that students add entries to their logbooks each week, and submit their logbooks for assessment quarterly. For each of the acquired skills reported quarterly, students must provide a date, rating, and example of something they have said or done that can be used, in part, as evidence that they possess the selected skill.

Robotics 801A provides numerous opportunities for students to communicate ideas, plans, and results of experimentation. Students are expected to use a variety of modes of representation (numeric, symbolic, graphical, linguistic) to communicate effectively and efficiently. The Learning Journal should act as the means of communicating their ideas/plans and reflecting on past modes of representation.

Student Reflection, Skills Logbook

Tasks for Instruction and/or Assessment

Portfolio

- Identify skills in your logbook that you feel you have developed. Identify the dates that you worked on the outcome and rate you technical skill at this time. Use your Evidence of Learning worksheets and Learning Journal to provide evidence of learning or developing this skill. Identify something you have said or done that can be used, in part, as evidence that you possess that skill. Notes, drawings, and comments may also be used to demonstrate the skill (117-9)

Paper and Pencil

- Record in your Learning Journal ideas and plans using several modes of communication (numeric, symbolic, graphical, linguistic). (215-2)

- Reflect on the entries in your logbook. Were there entries that communicated ideas more effectively than others? Explain. (215-2)

Journal

- Evaluate your, and your group's, effectiveness in planning, problem solving, decision making, and completing a task. Summarize your team's overall effectiveness, and identify ways to work more effectively. (215-7)

Interview

- Conduct an interview with your teacher to evaluate your self- and peer-assessments. Discuss the strengths and identify strategies for personal growth. (215-7)

Resources/Notes

Appendix C: Skills Logbook

Technical Skills Rating
Skills Logbook
Evidence of Learning
Learning Journal
Assessment Rubric

Introduction to Autonomous Robotics
~3 Classes

Outcomes

Students will be expected to

- ***explain the similarities and differences between radio-controlled and autonomous robots (300-1)***
- ***provide examples of how science and technology are an integral part of their lives and their community (117-5)***
- ***identify and describe the basic components of radio-controlled and autonomous robotics subsystems (300-2)***
- ***analyse how robotics subsystems interact (300-3)***

Elaborations—Strategies for Learning and Teaching

As an introduction, student groups can be given a variety of pictures and examples of radio-controlled (RC) and autonomous robots. Students will be familiar with the function of RC robots from their prior experiences in Applied Science 701A. However, the concept of autonomous robots will be new. The group task would be to review the pictures and identify the similarities and differences between RC and autonomous robot designs and functions. The groups can then create an operational definition of an autonomous robot which can be shared with the class.

Each part involved in the construction of a robot falls into one of six categories, or subsystems. The subsystems include structure, motion, power, sensor, control, and logic. Students are expected to identify and describe the six categories of robot subsystems and the parts associated with each. To assist in addressing this outcome and in managing resources, student teams could create a parts inventory. This activity may also help clarify the functions of the individual components of the robot subsystems. It is vital that students understand how the subsystems interact before they can appreciate how programming can be used to allow a robot to complete a task.

Introduction to Autonomous Robotics

~3 Classes

Tasks for Instruction and/or Assessment

Performance / Journal

- Create an inventory of the robot components provided by your teacher by first organizing the components into subsystem categories. In your learning journal, briefly describe the function of each component. (300-2)

Journal

- Reflect on how radio-controlled and autonomous robots could be used to improve an aspect of your life, or society in general. What do you think would be some of the advantages or disadvantages of using these robotics technologies? (117-5)

Paper and Pencil

- Given a variety of devices, or photographs of devices, explain which you would consider to be an autonomous robot, and which you would not. (300-1)

- Analyse using notes, comments, and drawings how robot subsystems interact. (300-3)

- Provide examples of how radio-controlled and autonomous robots are used in your community. What are some issues (e.g., cost, effectiveness, maintenance) with integrating this technology into society? (117-5)

Resources/Notes

VEX Inventor's Guide

Drivetrain, Microcontroller, and Transmitter

~4 Classes

Outcomes

Students will be expected to

- ***collaborate with team members to construct and test a squarebot robot, and troubleshoot problems as they arise. (214-14/215-6a)***
- ***configure a transmitter and microcontroller effectively and accurately to control a robot (213-3a)***
- ***evaluate the design and function of a squarebot robot in terms of stability, propulsion, speed, and control (118-4a)***

Elaborations—Strategies for Learning and Teaching

This is the first opportunity in the course for students to construct a robot. This build provides teachers with an opportunity to assess students' levels of prior knowledge and familiarity with the VEX robotics systems. Students should be provided with explicit instructions and diagrams to assist in the construction, so that they have exposure to proper design techniques. Furthermore, this guided activity will assist in building student confidence in constructing future autonomous robots.

The first robot construction (VEX squarebot robot) will consist of a basic square chassis design with propulsion involving one motor per side. Constructing this basic robot should provide students with an opportunity to review skills acquired in the course prerequisite, Applied Science 701A. These include fabrication, communication, assigning a group task, sharing team responsibilities, and troubleshooting problems as they arise.

The squarebot robot can be used in subsequent curriculum sections involving autonomous robot sensors and programs. Students should configure their robot to operate with arcade-stye propulsion and then with tank-style (double joystick controls) propulsion using the same motor configuration. This review of Applied Science 701A prerequisite knowledge will improve students' ability to configure the microcontroller and transmitter, skills that will be required in the autonomous programming of their robots. Students should also evaluate the squarebot design in terms of stability, propulsion, speed, and control. Students will be required to describe and evaluate the advantages and disadvantages of this design.

Please Note: It is very easy for students to mistakenly use VEX Servos instead of VEX Motors, as they look identical. It is recommended that all VEX Servos be labelled with bright permanent markings.

Drivetrain, Microcontroller, and Transmitter

~4 Classes

Tasks for Instruction and/or Assessment

Performance

- Construct a basic radio-controlled robot consisting of a square chassis design with one motor per side. (214-14/215-6a)

- Configure a transmitter to function in arcade-style and in tank-style. (213-3a)

Journal

- Evaluate your, and your group's, effectiveness in planning, problem solving, decision making, and completing a task. Identify your, and your team's, strengths, and define areas in which you and your team can be more effective. If you were going to do this build again, what would you do the same and what would you do differently? (215-7)

Paper and Pencil

- Describe the difference between arcade- and tank-style propulsion. Evaluate these configuration styles by first creating a list of advantages and disadvantages of each. Are there some situations where one style would be preferred over the other? Explain. (118-4a)

- What are the advantages and disadvantages of the squarebot design over other chassis that you have built? Evaluate this design in terms of stability, propulsion, speed, and control. (118-4a)

- Describe how to configure the transmitter and microcontroller to control different robotic functions. (213-3a)

Resources/Notes

VEX Squarebot Robot

Compound Gears
~7 Classes

Outcomes

Students will be expected to

- ***qualitatively describe and quantitatively calculate the gear ratios of compound gear trains (300-5)***
- ***design and construct a device that manipulates the relationship between gearing, speed, and torque (212-3a)***
- ***collaborate with team members to develop and implement a plan, and troubleshoot problems as they arise (215-6)***

Elaborations—Strategies for Learning and Teaching

In Applied Science 701A students learned about various gear types (e.g., spur, bevel, worm, rack, idler) and how they could be arranged to transfer power (e.g., rack and pinion, differential) or create a mechanical advantage (e.g., gear ratios). The purpose of this section is for students to expand on the knowledge that they have learned on simple gears to create a compound gear arrangement (e.g., gearbox). Students should be reminded to keep careful notes in their learning journals. The gear arrangements created here could later be modified for the autonomous robot challenges at the end of the course.

Students should be instructed that when more than one pair of simple gears are used together, the arrangement is called a compound gear train. The gear ratios for each indvidual gear pair are multiplied together to calculate the overall compound gear ratio for the gear train. Teachers should also use this time to revisit the concepts of free speed and stall torque as it applies to motor performance. Students should be reminded how gear ratio affects speed and torque. For a set motor power, a gear ratio that increases speed will decrease torque, and vice versa.

Torque = Force x Radius

Power = Force x Speed

$$\text{Gear Ratio} = \frac{\text{\# of Teeth on Driven Gear}}{\text{\# of Teeth on Driving Gear}}$$

Compound Gear Ratio = Gear Ratio 1 x Gear Ratio 2 x ....

Students should be provided with an opportunity in their groups to construct a radio controlled device to qualitatively and/or quantitatively measure the effects of a compound gear train. Suggestions may include a challenge to lift a predetermined load, or a tug of war challenge (modifications could be made to the squarebot chassis built earlier). Instead of providing explicit instructions for students to follow, teachers may wish to use this opportunity to allow students more choice in their design and build. This open, student-directed approach is more in keeping with the final autonomous robot challenges that are required in the latter part of the course. Using a familiar topic (gears) as an introduction to open builds may provide teachers and students with an easier transition from guided to open design.

Compound Gears

~7 Classes

Tasks for Instruction and/or Assessment

Performance

- Construct a gearbox to demonstrate several different gear ratio combinations. (300-5, 215-6, 212-3a)

Journal

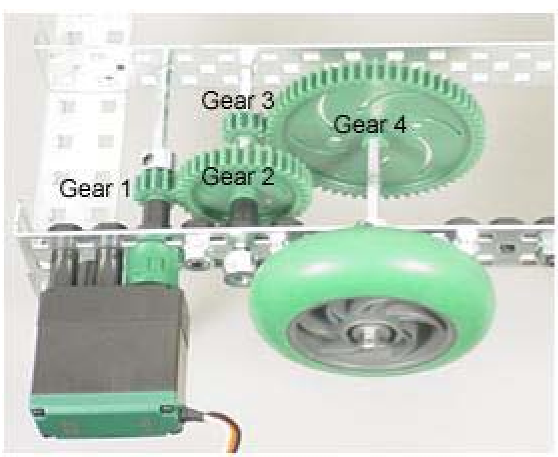
- In your learning journal, demonstrate an understanding of some robotic applications that you think would benefit from the use of a compound gear train. (300-5)

- How would you explain to a grade 6 student the gear ratio tradeoff between speed and torque? (300-5)

Paper and Pencil

- Describe the advantages and disadvantages of using a compound gear train. (300-5)

- Determine the compound gear ratio for the complex gear train shown below.
  Gear 1: 12 teeth
  Gear 2: 36 teeth
  Gear 3: 12 teeth
  Gear 4: 60 teeth (300-5)



Resources/Notes

Internet: How Do Gear Ratios Affect Speed and Torque http://www.education.rec.ri.cmu.edu/roboticscurriculum/vex_online/lessons/gearbox/lesson.html

VEX Inventor's Guide, pp. 3-5 to 3-8

Centre of Gravity and Degrees of Freedom
~7 Classes

Outcomes

Students will be expected to

- ***analyse how the stability and dexterity of a robotic manipulator are affected by its centre of gravity and degrees of freedom (300-6)***
- ***determine a robot's centre of gravity (212-3b)***
- ***collaborate with team members to construct and test a robotic manipulator, and troubleshoot problems as they arise (214-14/215-6 b)***

Elaborations—Strategies for Learning and Teaching

Object manipulation is one of the primary objectives in most RC and autonomous robotic developments today. Students were exposed to various types of object manipulators (e.g., plow, scoop, grabber) in Applied Science 701A and have evaluated each type based on its design, purpose, and effectiveness. Two important concepts that affect a robot's ability to manipulate an object are its stability and dexterity.

The stability of a mobile robot will be greatly affected by its centre of gravity. The centre of gravity is an average position of all the weight on the robot. This is important when considering forces acting on a robot, especially a robot in motion. Although students are not required to determine their robot's centre of gravity mathematically, they should be able to determine it experimentally.

A robot's dexterity will be a function of the number of degrees of freedom that it contains. A degree of freedom is the ability to move in a single direction of motion. To be able to move in multiple directions (e.g., to move both up and down and right to left) means to have multiple degrees of freedom. The human arm, and other examples from nature, should be used to reinforce the concept of degrees of freedom.

Students should be encouraged to research current uses of robotic manipulators and how they address centre of gravity and degrees of freedom considerations. Modifications of existing technologies can be modelled using VEX components or other methods. Teachers should encourage students to evaluate various designs to determine which is best for their robot.

The lesson should lead to a culminating activity where students are challenged to design and build a robot that can manipulate a predetermined object (e.g., lift an object off a shelf, open a pop bottle, stack cans, turn a book's page). In designing their robot's manipulator and frame, students will also have to consider the object's initial orientation, final orientation, shape, and mass. This will in turn affect the design of the robot and manipulator in terms of their required size, strength, grip, and gear reduction.

Teachers should remind students to keep accurate notes of their design and evidence of learning in their logbooks. Students may need to revisit their notes when considering how to complete the open-ended challenges later in the course.

Centre of Gravity and Degrees of Freedom
~7 Classes

Tasks for Instruction and/or Assessment

Performance

- With your team, assemble a mobile robotic manipulator that can accomplish a predeterined task. Identify how your group's design accounts for centre of gravity and degrees of freedom considerations. (214-14/215-6 b)

- Determine the centre of gravity of your robotic manipulator. (212-3b)

Observation / Journal

- Evaluate a manipulator that was designed by another team in your class. Critique the manipulator's strengths and weaknesses and summarize how you would improve their design. Record these observations in your logbook and apply them to your open-ended challenges at the end of the course. (300-6)

Paper and Pencil

- How does the location, elevation, and length of a manipulator affect the centre of gravity of the robot? (300-6)

- Summarize how the object to be manipulated will affect the design of the manipulator in terms of its required strength, grip, and gear ratio. (300-6)

- Analyse examples of object manipulation performed by animals in the natural world. How have they adapted for centre of gravity and degrees of freedom considerations? Could you transfer their adaptations to your group's design? (300-6)

Resources/Notes

VEX Inventor's Guide, pp. 2-27 to 2-31

Internet: Centre of Gravity http://www.education.rec.ri.cmu.edu/roboticscurriculum/vex_online/lessons/mechanics/center_of_mass/mechanics_center_of_mass.html

Unit 13: pp. 1-11

Introduction to Sensors

~3 Classes

Outcomes

Students will be expected to

- ***describe the functioning of domestic and industrial autonomous robotic technologies that are an integral part of our lives and community (116-5/117-5)***
- ***explain how the sensor subsytem functions and how it interacts with other robot subsystems***
  - input vs. output (300-7)
- ***define the difference between analog and digital sensors and specify applications for each (300-8)***

Elaborations—Strategies for Learning and Teaching

At the beginning of the course, a suggested teaching strategy was to provide students with images of radio-controlled and autonomous robots to help students establish a working definition of autonomous robotic technology. Teachers could use these images of autonomous robots again and ask students to discuss/describe the robots' functions and behaviour. Domestic and industrial examples should be provided. In essence, students should be able to identify what the robot has been designed to do. Teachers should limit the discussion to sensors, inputs and outputs. Programming will be discussed in later sections.

An autonomous robot can gather information about its environment and adjust its own behaviour based on that knowledge. The sensors are the "eyes and ears" of the robot. A sensor will generally tell the robot about one very simple thing in the environment (input) and the robot's program will interpret that information to determine how it should react (output).

A sensor is designed to measure a physical quantity such as sound, light, pressure, or rotation and convert it to an electronic signal of some kind (e.g., voltage). Sensors communicate this information in one of two ways: as an analog signal or as a digital signal. Analog sensors (e.g., light sensor, line following sensor, potentiometer) send continuous signals of any value between a minimum and maximum voltage. Digital sensors (e.g., bumper switch, limit switch, optical shaft encoder, ultrasonic sensor) send discrete signals of only the minimum or maximum voltage.

The weakness of analog sensors is that it is very difficult to send and maintain an exact, specific voltage. Sometimes an analog signal can be lost or interpreted as "noise" which can lead to erratic behaviour. The weakness of a digital sensor is that it only collects and sends information in discrete intervals and cannot detect a range of values.

Teachers should identify different scenarios where a digital sensor would be preferred over an analog sensor, and vice versa. The specific applications of each of the seven VEX sensors will be described in the next section.

Introduction to Sensors

~3 Classes

Tasks for Instruction and/or Assessment

Performance

- Research various examples of domestic and industrial autonomous robots. For each example, describe the robot's behaviour and identify what sensors it uses to collect information about its environment. (116-5/117-5)

Journal

- Is there any difference in the way that an autonomous robot senses information about its surroundings and the way that you sense information about your surroundings? (300-7)

- What limitations do autonomous robots have over humans in the way that they sense information from their environment? What advantages do they have over humans? (300-7, 116-5/117-5)

Paper and Pencil

- What sensors does your body have that tell you about your environment? Provide some examples of how information from these sensors affect your behaviour. (300-7)

- Provide an example of an autonomous robot and describe its behaviour in terms of inputs and outputs. (300-7)

- What is the difference between an analog and digital sensor? What are the main advantages and disadvantages of each type? (300-8)

Resources/Notes

VEX Inventor's Guide, p. 5-2, pp. 5-5 to 5-13

Internet: Sensors

http://www.education.rec.ri.cmu.edu/roboticscurriculum/vex_online/programming/sensors/index.htm

Introduction to Sensors (continued...)

~3 Classes

Outcomes

Students will be expected to

- ***identify and describe the purpose and function of each VEX sensor***
  - bumper switch
  - limit switch
  - light sensor
  - ultrasonic sensor
  - optical shaft encoder
  - potentiometer
  - line follower (300-9)

- ***analyse a given scenario and determine what sensors could be used to accomplish the task (300-10)***

- ***construct arguments to support a decision or judgement, using examples and evidence and recognizing various perspectives (118-6)***

- ***communicate questions, ideas, and intentions, and receive, interpret, understand, support, and respond to the ideas of others (215-1)***

- ***identify multiple perspectives that influence a science-related decision or issue (215-4)***

Elaborations—Strategies for Learning and Teaching

Teachers should identify and describe each of the seven VEX sensors and explain their purpose and function. The programming component should not be covered at this time. Students should be able to identify each sensor as either analog or digital and describe how each sensor gathers information.

The bumper and limit switch sensors are mechanical devices used to make or break connections within an electrical circuit. They are most often used to stop or start the motion of the robot or of a component on the robot. These digital sensors are physical switches and can only be activated, or triggered, by contact with an object. When the trigger is removed, the switch springs back to its natural state and there is no signal sent to the microcontroller.

The light sensor and line follower are both analog optical sensors that gather information on light intensity. The light sensor can be used to follow a beam of light or avoid shadows. The line follower can be used to follow a marked path, or discern a boundary between two high-contrast surfaces.

The ultrasonic sensor uses sonar to navigate and detect obstacles. Students should understand that this digital sensor is unique in that it has both an input and output signal. The emitter creates a high frequency sound pulse (output). The detector receives the echo of this pulse (input) after it reflects off of an obstacle.

The digital optical shaft encoder and analog potentiometer are both used to determine position. The potentiometer can determine angular position which can be very helpful in object manipulation. The optical shaft encoder is able to accurately measure the robot's direction and position regardless of the robot's motor power, gear ratio, or speed.

Additonal documentation on each VEX sensor's purpose and function can be found in the Sensors section of the Inventor's Guide or online at http://www.vexrobotics.com/.

Teachers should present the class with three or four different scenarios that describe tasks that are to be completed by an autonomous robot. In their groups, students would have to analyse the scenario and identify what sensor is needed to accomplish each task and explain why. Students should be challenged to think of how a different sensor could be used to accomplish the same task. They should also evaluate and critique the solutions proposed by other teams.

Introduction to Sensors (continued...)
~3 Classes

Tasks for Instruction and/or Assessment

Performance

- In your group, analyse a given scenario and identify a sensor that could be used to meet the predetermined task. Present your group's sensor to the class and explain why you feel it is the best choice. Evaluate and respond to the choices presented by other groups. Have other groups' perspectives influenced your decision as to what sensor is best suited to meet the task? (118-6, 215-1, 215-4, 300-10)

Paper and Pencil

- Explain how each of the seven autonomous sensors function. (300-9)

- Could some the analog sensors operate as digital sensors? If so, what advantages/disadvantages would they now have? (300-8, 300-9)

- Explain how a specific sensor could be used to stop a robot one metre away from a wall. Could a different sensor be used to replicate the same behaviour? If so, which one and how? (300-9, 300-10)

Journal

- In your learning journal, describe what sensors would be needed for an autonomous robot to pick up a piece of litter and place it in a garbage can. (300-9, 300-10)

Resources/Notes

VEX Inventor's Guide: Sensors

Internet: VEX website
http://www.vexrobotics.com/

Introduction to Programming

~6 Classes

Outcomes

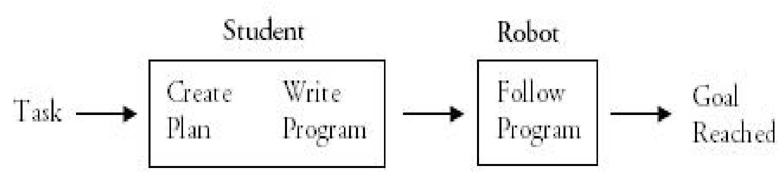
Students will be expected to

- ***explain programming (300-11)***
- ***identify and describe programming syntax and keywords required to write a program (300-12)***

Elaborations—Strategies for Learning and Teaching

Students should now have an appreciation of how sensors work, and the types of sensors available to be used on their robots. At this time, teachers should begin to explain to their students that robots are programmed to complete a function. Teachers should define a program as a sequence of instructions and steps that the robot can interpret and carry out to accomplish a goal. A program may be a set of instructions that do not require any input from the outside environment. However, the functionality and "intelligence" of the robot can be greatly increased by creating a program that relies on outside information gathered through sensors and interpreted by the robot (microcontroller). The use of a program that relies on sensors will allow an autonomous robot to change its behaviour depending on the conditions of its environment.

To accomplish a goal, the student programmer must come up with the plan and communicate it to the robot. The robot must follow the plan. Because the students and the robot do not speak the same language, students must learn how to write their plans in programming language. Just like spoken languages, each programming language has its own syntax. Syntax is the vocabulary, grammar, word-choice, and structure that make up the programming language. Teachers should explain to the students the keywords that they will use to program their robot (e.g., IF, ELSE, WHILE, WAIT).



Introduction to Programming
~6 Classes

Tasks for Instruction and/or Assessment

Paper and Pencil

- Define programming. What is the purpose of writing a program? (300-11)
- Identify some of the keywords used in programming syntax and describe what their functions are in the program. (300-12)

Resources/Notes

Internet: Thinking about Programming
http://www.robotc.net/education/curriculum/vex/pdfs/Fundamentals%20-%20Thinking%20about%20Programming.pdf

Internet: Programming Logic
http://www.education.rec.ri.cmu.edu/roboticscurriculum/vex_online/programming/robotc/fundamentals.html

Introduction to Programming *(continued...)*
~6 Classes

Outcomes

Students will be expected to

- ***create flowcharts involving sub-programs to break a complicated behaviour into a series of simple tasks (300-13)***

Elaborations—Strategies for Learning and Teaching

Students must understand that an important idea in behaviours is that they can be built up or broken down into other behaviours. Complex behaviours, like going through a maze, can always be broken down into smaller, simpler behaviours. These simple or basic behaviours can then be programmed into the robot. In this way, analysing a complex behaviour maps out the pieces that need to be programmed, then allows the student to program them, and put them together to build the code that the autonomous robot will follow to reach the goal.

Students should be given opportunities to practise breaking a complex behaviour (e.g., walking out of a classroom, brushing their teeth, opening a pop bottle) into smaller, simpler tasks. Teachers should encourage groups of students to create and present flowcharts that outline the progression of each simple task needed to reach the overall goal. Other groups of students could evaluate (debug) the presented flowcharts. The teacher could then present the class with a flowchart of a class program and have the students/groups act as sub-programs. The entire class could run through the program and troubleshoot any errors.

Once students have experience with the principles of programming and how to break a complex behaviour into a series of simple tasks, it is time to introduce them to the software that they will use to program their robots - easyC Pro. The easyC Pro software allows students to represent each simple task with an easyC Pro icon. These icons model the simple task behaviors and are written in a programming language that the robot can understand. Organizing these icons in the proper order (e.g., as they appear in the students' flowcharts) recombines the simple tasks into one complex behaviour that the robot can follow. After the completed program is downloaded into the VEX microcontroller, the robot can run the program and meet the desired goal.

Introduction to Programming (continued...)
~6 Classes

Tasks for Instruction and/or Assessment

Performance

- In your group, create a flowchart that breaks a complex behaviour (e.g., walking out of the classroom, brushing your teeth, opening a pop bottle) into smaller, simpler tasks. Once you have completed your flowchart present your program to the rest of the class. Have one member of your group act as the "robot" and follow the program as it is read out loud by another group member.

  Evaluate the programs that are presented by other groups. Do you see any errors in their programming? Are there any steps that were overlooked or steps that could be better planned? (300-13)

- As an entire class reenact a program that is presented by the teacher. Different students should represent different tasks in the flowchart. Run the program. Are there any parts of the flowchart that are missing. Are there any complex behaviours that could be broken down into simpler tasks. Evaluate the program and make recommendations on how it could be better planned. Try out the revised program to see if the goal could be reached in fewer/ simpler steps. (300-13)

Resources/Notes

Internet: Thinking about Programming
http://www.robotc.net/education/curriculum/vex/pdfs/Fundamentals%20-%20Thinking%20about%20Programming.pdf

Internet: Programming Logic
http://www.education.rec.ri.cmu.edu/roboticscurriculum/vex_online/programming/robotc/fundamentals.html

Introduction to Programming *(continued...)*
~6 Classes

Outcomes

Students will be expected to

- ***manipulate and use program flow statements and functions to program a robot to complete a task (300-14)***
- ***diagnose the different inputs and outputs that can be connected to the robot logic subsystem (300-15)***
- ***program a robot to operate in radio controlled, autonomous, and combined setups (300-16)***
- ***collaborate with team members to develop and carry out a plan, and troubleshoot problems as they arise (215-6)***
- ***evaluate individual and group processes used in planning, problem solving and decision making, and completing a task (215-7)***

Elaborations—Strategies for Learning and Teaching

The ***VEX Inventor's Guide*** Programming section (pp. 8-19 to 8-58) should be used to guide students through their first experience in programming with easyC Pro. This step-by-step guided tutorial progresses students through the fundamentals of programming. They will learn to use the online window to control and monitor the analog inputs, the digital inputs/outputs, and the motors that are connected to the robot's microcontroller. The online window is a valuable tool for students to become more familiar with sensor and motor functions and control.

The programming guide can be used to teach students proper programming sequence and how to program motors and sensors in radio controlled, autonomous, and combined setups. Students will need to become familiar with these three setups in order to complete the guided sensor builds and open-ended challenges that appear later in the course.

Teachers should take time during this section to identify any strengths or weaknesses that their students/groups may have. The teacher should facilitate each group's progress but restrain from providing direct instruction. Since students will have no prior experience with the easyC Pro software, this is an opportune time for groups to work collaboratively to problem solve and troubleshoot on their own. Group members will need to learn from each other and begin to work collectively as a team. Teachers should ensure that all students have experience in programming and an opportunity to contribute to the group.

The guided sensor builds outlined in subsequent sections should only be covered after students have met the curriculum outcomes in the Introduction to Sensors and Introduction to Programming sections.

Please note: the easyC Pro software has an extensive help section that may be a valuable resource to both teachers and students. There are also easyC programming tutorials available online at www.youtube.com.

Introduction to Programming *(continued...)*
~6 Classes

Tasks for Instruction and/or Assessment

Performance

- In your group, complete the Programming section in the ***VEX Inventor's Guide***. Ensure that each group member has an opportunity to participate and contribute to the team. (300-14, 300-16, 215-6)

Paper and Pencil

- Explain how the online window in easyC Pro can be used to carefully monitor and control sensors and motors. (300-15)
- What are some robot behaviours or operations that would benefit from a radio controlled setup? From an autonomous setup? From a combined setup? (300-16)

Journal

- Reflect on how well your group worked together on completing the programming section in the ***VEX Inventor's Guide***. Are there any areas that you, or other group members, need to improve on to work better as a team? If so, what are they and how do you plan to address them? (215-6, 215-7)

Resources/Notes

VEX Inventor's Guide, pp. 8-19 to 8-58

easyC Pro help section

Internet: easyC online tutorials
http://www.youtube.com/watch?v=ZC_IzM9frGo

Autonomous Guided Builds
~15 Classes

Outcomes

Students will be expected to

- ***carry out procedures controlling the major variables and adapting or extending procedures where required (213-2)***

Elaborations—Strategies for Learning and Teaching

By making good use of sensors to detect the important aspects of its environment, a robot can operate completely independent (autonomous) of human control. Students should now begin to program robots with each type of VEX sensors to complete a predetermined task. They will follow guided programming and build challenges to become more familiar with the capabilities and limitations of the easyC Pro program and the VEX sensors. Students should then use their knowledge of sensors and programming features to create radio controlled robots with autonomous features in the open-build competitions at the end of the course.

After each guided challenge, the teacher could assess each student or groups of students by asking them to make a slight modification to the program that would change the robot's performance. Or, teachers could provide students with a handout of a similar program to the one they've created and ask them to describe or troubleshoot the program. This activity is to be used to obtain some individual assessment and also to bridge the gap between the guided inquiry outcomes of this section with the open inquiry outcomes of the final project.

- ***use contact sensors effectively and accurately for collecting data to assist in the control of robot, or robot component, movement (213-3b)***
- ***collaborate with team members to construct and test a robot with autonomous features, and trobleshoot problems as they arise (214-14/215-6 c)***

Students should conduct a guided build that incorporates a digital contact sensor (limit switch or bumper switch). These sensors are physical switches that can only be activated through contact with an object. Once the switch is activated, a signal is sent to the robot to activate or deactivate a prescribed action. An example program could involve a limit switch to detect the edge of a table or to use bumper switches to detect contact with a wall or other object.

Students must work co-operatively in their groups to carry out a procedure, gather the required materials, build the robot, and program it using easyC Pro software. Teachers should ensure that all group members have an opportunity to participate and contribute to their group's activities. Students will have to work together to troubleshoot problems as they occur and adapt or extend the existing procedures to meet the desired outcomes.

- ***identify new applications of a contact sensor that arise from what was learned (214-17a)***

Teachers should remind students to keep accurate notes of their contact sensor design and record evidence of learning in their logbooks. Students should revisit their notes when considering how to incorporate contact sensors in the open-ended challenges at the end of the course and in the creation of their presentations.

Autonomous Guided Builds

~15 Classes

Tasks for Instruction and/or Assessment

Journal

- Identify how limit and bumper switches can be incorporated into your final project robot. In your skills logbook, explain the purpose for the inclusion of these switches in your robot design. Determine how to incorporate the contact sensors in your robot for the open-ended challenge and configure the program to allow the switches to function as intended. (214-17a, 300-9)

Performance

- With your team, construct, program, and test a robot that can use contact sensors to accomplish a predetermined task. (213-3b, 214-14/215-6c)
- Adapt your program to combine radio controlled code with autonomous code. Extend the program so that your robot will drive in normal RC mode until one of the limit/bumper switches is pressed. When the contact sensor is pressed an autonomous code should automatically run. Once the sensor is no longer pressed, the robot will again operate in RC. (213-2, 213-3b)

Paper and Pencil

- Explain how easyC Pro code uses contact sensors information to control motor outputs. (213-3b, 300-9)
- Could the contact sensors be used to autonomously control other functions of the robot, besides motors? If so, how? (300-9, 214-17a)
- Given a printout of a program, make modifications to the program and predict the resulting robot behaviour. (213-2)

Resources/Notes

Student Resource: Autonomous Guided Builds - The Cliffhanger

VEX Inventor's Guide: Sensor Section

Autonomous Guided Builds (continued...)

~15 Classes

Outcomes

Students will be expected to

- ***use a light sensor effectively and accurately for collecting data to assist in the control of robot, or robot component, movement (213-3c)***
- ***identify new applications of an optical sensor that arise from what was learned (214-17b)***

Elaborations—Strategies for Learning and Teaching

Students should conduct a guided build that incorporates an analog optical sensor (light sensor or line follower sensor). These sensors detect light intensity and send this information to the microcontroller. The robot then responds to the light intensity in accordance to the program's directions. An example program could involve using a light sensor to move a robot towards a light source or away from shadows. The line follower sensor can be used to program the robot to follow a marked path (e.g., black line on a white surface).

Teachers should remind students to keep accurate notes of their design involving optical sensors and record evidence of learning in their logbooks. Students should revisit their notes when considering how to incorporate a light sensor or line follower sensor in the open-ended challenges at the end of the course and in the creation of their presentations.

Please note: the programming code for the line follower sensor can be quite complicated. The three-line follower program has the best accuracy and control, but requires the student to import libraries in easyC Pro and be familiar with the use of thresholds, Program Globals, and User Functions. The single-line follower code requires less programming but has lower accuracy and control. It is recommended that the line follower activity only be used as an enrichment activity (if time allows) or as a higher-ended challenge for experienced easyC Pro users. The advanced tutorials in easyC Pro can be used as a starting point for interested students.

Autonomous Guided Builds *(continued...)*

~15 Classes

Tasks for Instruction and/or Assessment

Journal

- Identify how optical sensors can be incorporated into your final project robot. In your skills logbook, explain the purpose for the inclusion of these sensors in your robot design. Determine how to incorporate the optical sensors in your robot for the open-ended challenge and configure the program to allow the sensors to function as intended. (214-17b, 300-9)

Performance

- With your team, construct, program, and test a robot that uses a light sensor to accomplish a predetermined task. (213-3c, 214-14/215-6c)

- Adapt your program to minimize the amount of tracking/overshoot that the robot experiences as it follows the light source. Refine the motor drive times and light sensor scan times to optimize your robot's performance. Once you have modified the programming code, challenge another team to see whose autonomous robot can get to the light source first. (213-2, 213-3c)

Paper and Pencil

- Explain how easyC Pro code uses optical sensors information (light sensor and line follower) to control motor outputs. (213-3c, 300-9)

- Could the optical sensors be used to autonomously control other functions of the robot, besides motors? If so, how? (300-9, 214-17b)

- Given a printout of a program, make modifications to the program and predict the resulting robot behaviour. (213-2)

Resources/Notes

Student Resource:
Autonomous Guided Builds - Light Sensor

VEX Inventor's Guide: Sensor Section

easyC Pro help section:
- Advanced Tutorials

Autonomous Guided Builds *(continued...)*
~15 Classes

Outcomes

Students will be expected to

- ***use an ultrasonic sensor effectively and accurately for collecting data to assist in the control of robot, or robot component, movement (213-3d)***
- ***use a position sensor effectively and accurately for collecting data to assist in the control of robot, or robot component, movement (213-3e)***
- ***identify new applications of an ultrasonic sensor that arise from what was learned (214-17c)***
- ***identify new applications of a position sensor that arise from what was learned (214-17d)***

Elaborations—Strategies for Learning and Teaching

Students should conduct a guided build that incorporates a digital ultrasonic sensor. This sensor emits a sonar pulse to detect how far the robot is from an object. An example program could involve an ultrasonic sensor to avoid obstacles in the robot's path (e.g., drive around barriers, back away from barriers) or to activate a robot function at a set distance (e.g., close/lift a manipulator when the robot is a set distance from the object to be manipulated).

Students should conduct guided builds that incorporate the features of position sensors (digital optical shaft encoder and analog potentiometer). These sensors detect linear or angular position. (The position reading can be interpreted by the microcontroller to move a manipulator to a desired height/angle or to move the robot to a desired area.) An example program could involve a potentiometer to lift an object to an exact height or to use an optical shaft encoder to guide a robot through a maze. The precise nature of these sensors allow for autonomous movements that would be very difficult to replicate in normal RC operation.

Teachers should remind students to keep accurate notes of their position sensor and ultrasonic sensor designs and record evidence of learning in their logbooks. Students should revisit their notes when considering how to incorporate an optical shaft encoder, potentiometer, and/or ultrasonic sensor in the open-ended challenges at the end of the course and in the creation of their presentations.

Please note: the Autonomous Guided Build, "Up and Over" allows students to use the transmitter to remotely activate an autonomous function. To accomplish this, students will use a new command in the easyC Pro software called Rx Input. This combination of RC and autonomous function will be very useful in the final project. The Rx Input command can be used to activate an autonomous function or to deactivate the robot/robot functions. This feature may be useful as a safety "kill switch" if an autonomous function may potentially lead to dangerous operation during the open-ended challenges. This way a student, or teacher, could remotely deactivate a robot that suddenly acts in an unintended way.

Autonomous Guided Builds (continued...)
~15 Classes

Tasks for Instruction and/or Assessment

Journal

- Identify how ultrasonic sensors, optical shaft encoders, and potentiometers can be incorporated into your final project robot. In your skills logbook, explain the purpose for the inclusion of these sensors in your robot design. Determine how to incorporate the sensors in your robot for the open-ended challenge and configure the program to allow the sensors to function as intended. (214-17c, 214-17d, 300-9)

Performance

- With your team, construct, program, and test a robot that can use an ultrasonic sensor to accomplish a predetermined task. (213-3d, 214-14/215-6 c)
- Adapt your ultrasonic program so that instead of backing away from an obstacle, your robot drives around the obstacle and continues on its path. Work with your group to determine safe clearances, and how to deal with obstacles like walls. Once you have adapted the programming code, demonstrate the new function to your teacher and classmates. (213-2, 213-3d)
- With your team, construct, program, and test a robot that can use a position sensor to accomplish a predetermined task. (213-3e, 214-14/215-6 c)
- Adapt your program to change one of the aspects of the position sensor (e.g., program your robot to drive through the maze in reverse, program your robot to lift an object to a new height). Once you have adapted the programming code, demonstrate the new function to your teacher and classmates. (213-2, 213-3e)

Paper and Pencil

- Explain how easyC Pro code uses information from the ultrasonic sensor (inputs and outputs) to control motor outputs. (213-3d, 300-9)
- Explain how easyC Pro code uses information from the position sensors (optical shaft encoder and potentiometer) to control motor outputs. (213-3e, 300-9)
- Could the ultrasonic sensor and position sensors (optical shaft encoder and potentiometer) be used to autonomously control other functions of the robot, besides motors? If so, how? (300-9, 214-17c, 214-17d)

Resources/Notes

Student Resource:
Autonomous Guided Builds -
The Scaredybot
That's Amazething!
Up and Over

VEX Inventor's Guide: Sensor Section

Final Project

~30 Classes

Outcomes

Students will be expected to

- ***propose alternative solutions to a given practical problem, identify the potential strengths and weaknesses of each, and select one as the basis for a plan (214-15)***
- ***design and construct a robot with autonomous and RC functions (212-3c)***
- ***collaborate with team members to construct and test a robot design, using components and sensors conducive to completing a predetermined task, and troubleshooting problems as they arise (214-14/215-6 d)***
- ***select and use appropriate numeric, symbolic, graphical, and linguistic modes of representation to communicate ideas, plans, and results (215-2)***

Elaborations—Strategies for Learning and Teaching

These outcomes will be addressed as a group by having students engage in two culminating final projects that integrate their combined knowledge and skills obtained throughout this course. The first challenge will require student groups to design, construct, and program a RC robot with at least 1 autonomous sensor function. The second competition will require the incorporation of at least 3 autonomous sensor functions in the robot design. Student groups should be provided with the autonomy to create a unique design to meet the predetermined challenges. Teachers are encouraged to facilitate the process by having students reflect on the design principles they have learned and continuously evaluate their designs for effectiveness in accomplishing the prescribed robot tasks.

The final project will involve teams members working co-operatively to construct and test a robot design using components conducive to completing a predetermined task, and troubleshooting problems as they arise. This process will involve the consideration of alternative solutions and the identification of potential strengths and weaknesses of each.

Robots should be assessed based on their effectiveness in completing the assigned tasks. The design of the robot should be assessed in accordance to a component construction and sensor programming rubric (provided to students in advance). Furthermore, student teams should be assessed on the engineering design process.

In addition to the planning, construction, and robot competition, it is expected that student teams develop a slide show presentation that documents the engineering design process which they used. Students should be provided with the presentation criteria and rubric by which they will be assessed. The criteria should involve the benefits of the particular designs chosen (e.g., chassis, manipulators, sensors, etc.) as well as any pertinent calculations such as theoretical/actual speeds, gear ratios, maximum loads, etc..

Final Project
~30 Classes

Tasks for Instruction and/or Assessment

Performance

- Your team must reflect on all of the past mechanisms, subsystem designs, and sensor programming in order to create a robot to complete a predetermined task. There will be two separate competitions:
  A) The first competition will require your team to design, construct, and program a RC robot with at least 1 autonomous sensor function.
  B) The second competition will require your team to design, construct, and program a RC robot with at least 3 autonomous sensor functions.

  In addition to planning, constructing, and competing with your robot, your team will be responsible for developing a slide show presentation that documents the engineering design process used. The criteria for the presentation is outlined below.

  Competition Reporting Criteria
  You will be required to document (using an electronic presentation) all of the components of the engineering design process:
  1. Identify a need/challenge.
  2. Define a problem to be solved.
  3. Gather information.
  4. Conduct research.
  5. Find alternative solutions.
  6. Analyse the possible solutions.
     -chassis
     -propulsion analysis
     -mechanisms/sensors
  7. Design, test, and evaluate the best solutions.
     -chassis (center of gravity, sketches; pencil and/or Google SketchUp)
     -propulsion (drivetrain design, gear reduction calculations, sketches; pencil and/or Google SketchUp)
     -components (degrees of freedom, manipulators, sensors, programming, sketches; pencil and/or Google SketchUp)
  8. Build (illustrated via photograph).
  9. Communicate (using presentation software).

  (212-3c, 214-14d, 214-15, 215-6d)

Presentation

- Prepare and present a ten-minute presentation for your class on the engineering design process used for the construction of your robot. Include an introduction, conclusion, and references in addition to the components outlined by your teacher. (215-2)

Resources/Notes

Appendix A: RC/Autonomous Open Builds

Final Project Description

Final Project Rubrics
- Oral Presentation
- Engineering Design
- Component and Sensor Programming
- Robot Performance

Appendix A

RC/Autonomous Open Builds

RC / Autonomous Open-Build: Competition Teacher Handout

These outcomes will be addressed as a group by having students engage in two culminating final projects that integrate their combined knowledge and skills obtained throughout this course. Student groups should be provided with the autonomy to create a unique design to meet a predetermined challenge. Teachers are encouraged to facilitate the process by having students reflect on the design principles they have learned and continuously evaluate their design for effectiveness in accomplishing the prescribed robot task.

The final project will involve teams members working co-operatively to construct and test a robot design using components conducive to completing a predetermined task, and troubleshooting problems as they arise. This process will involve the consideration of alternative solutions and the identification of potential strengths and weaknesses of each.

Robots should be assessed based on their effectiveness in completing the assigned tasks. The design of the robot should be assessed in accordance to a chassis/component construction and sensor programming rubric (provided to students in advance). Furthermore, student teams should be assessed on the engineering design process.

In addition to the planning, construction, and robot competition, it is expected that student teams develop a slide-show presentation that documents the engineering design process which they used. Students should be provided with the presentation criteria and rubric by which they will be assessed. The criteria should involve the benefits of the particular designs chosen (e.g., chassis, manipulators, sensors, etc.) as well as any pertinent calculations such as theoretical/actual speeds, gear ratios, maximum loads, etc..

RC / Autonomous Open-Build: Competition Student Handout

Task:

Your team must reflect on all of the past mechanisms, subsystem designs, and sensor programming in order to create a robot to complete a predetermined task. There will be two separate competitions:

A) The first competition will require your team to design, construct, and program a RC robot with at least **1** autonomous sensor function.

B) The second competition will require your team to design, construct, and program a RC robot with at least **3** autonomous sensor functions.

In addition to the planning, construction, and competing with your robot, your team will be responsible for developing a slide show presentation that documents the engineering design process that you use. The criteria for the presentation is outlined below.

Competition Presentation Criteria:

You will be required to document (electronic presentation) all of the components of the engineering design process. Use the following statements to guide your presentation:

1. Identify a need/challenge.
2. Define the problem to be solved.
3. Gather information.
4. Conduct research.
5. Find alternative solutions.
6. Analyse the possible solutions.
    - -Chassis
    - -Propulsion analysis
    - -Mechanisms/Sensors
7. Design, test, and evaluate the best solutions
    - -Chassis (center of gravity, sketches; pencil and/or Google SketchUp)
    - -Propulsion (drivetrain design, gear reduction calculations, sketches; pencil and/or Google SketchUp)
    - -Components (manipulators, sensors, programming, sketches; pencil and/or Google SketchUp)
8. Build (illustrated via photograph)
9. Communicate (using presentation software)

Presentation Logistics and Assessment:

- Presentation to include an introduction, conclusion, and references
- Presentation duration (10 minutes; presentation and questions)
- Presentation to be assessed in accordance to the attached *Oral Presentation* rubric
- Engineering design process to be assessed in accordance to the attached *Engineering Design* rubric
- Robot design to be assessed in accordance to the attached *Competition A and B Robotics Components and Sensor Programming* rubric
- Robot performance to be assessed in accordance to the attached *Competition A and B Robotics Performance* rubric.

This is a unique opportunity to create a project of your own design... take pride in your work and enjoy!

Robotics 801A
FINAL PROJECT
ASSESSMENT RUBRIC: ORAL PRESENTATION

Categories	4 Marks	3 Marks	2 Marks	1 Mark	
Delivery	Speaker's voice is strong and clear. Speaker conveys confidence about the topic. Excellent eye contact.	Good speaking voice. Speaker is in command of topic. Good eye contact.	Clarity of speech is uneven. Speaker is not completely sure of topic or appears disengaged. Limited or sporadic eye contact.	Control of speaking tone, clarity, and volume is not evident. Speaker does not convey interest. No eye contact.	
Introduction Conclusion References	Introduction was clear with concise thesis statement and engaging. Conclusion tied presentation together and was memorable. All references were present.	Introduction was clear with concise thesis statement. Conclusion tied presentation together. All references were present.	Introduction not present or thesis statement unclear. Conclusion did not adequately summarize presentation. Missing references.	Introduction and /or conclusion missing or ineffective. Many references not present.	
Topic Coverage	Presentation clearly articulated the identified need, problem to be solved, evidence of other possible solutions considered, and support of the design of the robot components and sensors chosen.	Presentation contained the identified need, problem to be solved, evidence of other possible solutions considered, and support of the design of the robot components chosen; however, lack clarity in certain cases.	Presentation lacked clarity and key components such as one or two of the following: identified need, problem to be solved, evidence of other possible solutions considered, and support of the design of the robot components chosen.	Presentation lacked clarity and several of the following components: identified need, problem to be solved, evidence of other possible solutions considered, and support of the design of the robot components chosen.	
Content Accuracy	All content throughout the presentation is accurate. There are no factual errors.	Most of the content is accurate but there is one piece of information that may be inaccurate.	The content is generally accurate, but one piece of information is clearly flawed or inaccurate.	Content is typically confusing or contains more than one factual error.	
Use of Graphics	All graphics are attractive (size and colors) and support the theme/content of the presentation.	A few graphics are not attractive but all support the theme/content of the presentation.	All graphics are attractive but a few do not support the theme/content of the presentation.	Several graphics are unattractive and or ineffective and detract from the content of the presentation.	

Total: /20

Robotics 801A
FINAL PROJECT
ASSESSMENT RUBRIC: ENGINEERING DESIGN

Categories	4 Marks	3 Marks	2 Marks	1 Mark	
Need Identified / Problem Defined	Need for robot is clearly articulated and the problem(s) to be solved are clearly defined.	Need for robot is clearly articulated and the problem(s) to be solved is (are) not well defined.	Need for robot is not clearly articulated and the problem(s) to be solved is (are) not well defined.	Need for the robot is not clearly articulated and the problem(s) to be solved is (are) not defined.	
Chassis	Clear evidence exists of analysis of other chassis designs. Sketches (pencil or electronic) provided of the chassis design chosen. Explanation of chassis design chosen over other designs that were analysed was clearly articulated.	Some evidence exists of analysis of other chassis designs. Sketches (pencil or electronic) provided of the chassis design chosen. Explanation of chassis design chosen over other designs that were analysed was provided.	Little evidence exists of analysis of other chassis designs. Sketches (pencil or electronic) provided of the chassis design chosen. Explanation of chassis design chosen over other designs that were analysed was provided but unclear.	No evidence exists of analysis of other chassis designs. Sketches (pencil or electronic) provided of the chassis design chosen either not provided or unclear. Explanation of chassis design chosen over other designs that were analysed was not provided.	
Propulsion	Clear evidence exists of analysis of other propulsion designs. Sketches (pencil or electronic) provided of the propulsion design chosen. Explanation of propulsion design chosen over other designs that were analysed was clearly articulated. Gear reduction and theoretical speed calculations provided and accurate.	Some evidence exists of analysis of other propulsion designs. Sketches (pencil or electronic) provided of the propulsion design chosen. Explanation of propulsion design chosen over other designs that were analysed was provided. Gear reduction and theoretical speed calculations provided but with minor error.	Little evidence exists of analysis of other propulsion designs. Sketches (pencil or electronic) provided of the propulsion design chosen. Explanation of propulsion design chosen over other designs that were analysed was provided but unclear. Gear reduction and theoretical speed calculations provided but with minor error.	No evidence exists of analysis of other propulsion designs. Sketches (pencil or electronic) provided of the propulsion designs chosen either not provided or unclear. Explanation of propulsion design chosen over other designs that were analysed was not provided. Gear reduction and theoretical speed calculations either not provided or ineffective.	
Components and Sensors	Clear evidence exists of analysis of other component and sensor designs. Sketches (pencil or electronic) provided of the component and sensor design chosen. Explanation of component design chosen over other designs that were analysed was clearly articulated. Explanation of sensor(s) chosen over other sensor(s) that were analysed was clearly articulated.	Some evidence exists of analysis of other component and sensor designs. Sketches (pencil or electronic) provided of the component and sensor design chosen. Explanation of component design chosen over other designs that were analysed was provided. Explanation of sensor(s) chosen over other sensor(s) that were analysed was provided.	Little evidence exists of analysis of other component or sensor designs. Sketches (pencil or electronic) provided of the component and sensor design chosen. Explanation of component design chosen over other designs that were analysed was provided but unclear. Explanation of sensor(s) chosen over other sensor(s) that were analysed was provided but unclear.	No evidence exists of analysis of other component or sensor designs. Sketches (pencil or electronic) provided of the component and sensor designs chosen either not provided or unclear. Explanation of component design chosen over other designs that were analysed was not provided. Explanation of sensor(s) chosen over other sensor(s) that were analysed was not provided.	
Build and Programing	Illustration of the sequence of the build and programming process was excellent. Photographs taken were appropriate. Photograph quality was excellent.	Illustration of the sequence of the build and programming process was good. Photographs taken were appropriate. Photograph quality could have been improved.	Illustration of the sequence of the build and programming process needs improvement. Key photographs of the build process or steps in the programming process were missing. Photograph quality could have been improved.	Illustration of the sequence of the build and programming process was confusing. Large gaps in the programming process. Photographs taken provided little insight to the intricacy/complexity of the build process. Photograph quality was poor.	

Total: / 20

Robotics 801A
COMPETITION A
ASSESSMENT RUBRIC: ROBOTICS COMPONENTS AND SENSOR PROGRAMMING

Categories	4 Marks	3 Marks	2 Marks	1 Mark	
Chassis	Chassis is rigid and joints are square and secure. Chassis design is conducive to drivetrain and other components.	Chassis lacks some critical design components. Chassis is rigid and joints are square and secure. Chassis design is conducive to drivetrain and other components but could be more efficient in design.	Chassis lacks many critical design components. Chassis is relatively rigid and joints could be more square and secure. Chassis design is conducive to drivetrain and other components but could be more efficient in design.	Chassis lacks most critical design components. Chassis rigidity and joints require improvement The chassis design is not appropriately matched to the drivetrain or components. Components appear to be randomly attached.	
Drivetrain	Drivetrain design is excellent Drivetrain is securely attached to chassis. Gears are properly meshed and alignment is excellent. Design is conducive to functioning of robot.	Drivetrain lacks some critical design components. Drivetrain is securely attached to chassis. Gears are properly meshed and alignment is excellent. Some elements of the drivetrain are conducive to the functioning of robot.	Drivetrain lacks many critical design components. Drivetrain is securely attached to chassis. Gears could be more properly meshed and alignment is problematic. Few elements of the drivetrain are conducive to the functioning of robot.	Drivetrain lacks most critical design components. Drivetrain could be more securely attached to chassis. Gears could be more properly meshed and alignment is problematic.	
Object Manipulation and Degrees of Freedom	Object manipulator design is excellent. Robot has 2 or more degrees of freedom.	Object manipulator design is very good in general. Some elements of the design could be more conducive to the component purpose. Robot has more than 1 degree of freedom.	Object manipulator design is good. Many elements of the design could be more conducive to the component purpose. Robot has only 1 degree of freedom.	Object manipulator design needs improvement. Most elements of the design could be more conducive to the component purpose.	
Sensors and Programming	Program works flawlessly and is easy to understand and modify. Robot uses 1 or more autonomous sensor function.	Program works but is hard to understand or modify. Robot uses 1 autonomous sensor function.	Program and autonomous sensor do not work properly. Much more attention to the programming is required.	Program and autonomous sensor do not work.	
Overall	Robot design and programming are elegant. All components function smoothly, work well together, look like they belong together, and are easy to modify or replace. Efficient sensors and parts use.	Robot lacks some critical design components. Most components function smoothly, work well together, look like they belong together, and are easy to modify or replace. Efficient sensors and parts use.	Robot lacks many critical design components. Some components function smoothly, work well together, look like they belong together, and are easy to modify or replace. Parts and sensors use could have been more efficient.	Robot lacks most critical design components. Few components function smoothly, work well together, look like they belong together, and are easy to modify or replace. Parts and sensor use could have been more efficient.	

Total: / 20

Robotics 801A
COMPETITION B
ASSESSMENT RUBRIC: ROBOTICS COMPONENTS AND SENSOR PROGRAMMING

Categories	4 Marks	3 Marks	2 Marks	1 Mark	
Chassis	Chassis is rigid and joints are square and secure. Chassis design is conducive to drivetrain and other components.	Chassis lacks some critical design components. Chassis is rigid and joints are square and secure. Chassis design is conducive to drivetrain and other components but could be more efficient in design.	Chassis lacks many critical design components. Chassis is relatively rigid and joints could be more square and secure. Chassis design is conducive to drivetrain and other components but could be more efficient in design.	Chassis lacks most critical design components. Chassis rigidity and joints require improvement The chassis design is not appropriately matched to the drivetrain or components. Components appear to be randomly attached.	
Drivetrain	Drivetrain design is excellent Drivetrain is securely attached to chassis. Gears are properly meshed and alignment is excellent. Design is conducive to functioning of robot.	Drivetrain lacks some critical design components. Drivetrain is securely attached to chassis. Gears are properly meshed and alignment is excellent. Some elements of the drivetrain are conducive to the functioning of robot.	Drivetrain lacks many critical design components. Drivetrain is securely attached to chassis. Gears could be more properly meshed and alignment is problematic. Few elements of the drivetrain are conducive to the functioning of robot.	Drivetrain lacks most critical design components. Drivetrain could be more securely attached to chassis. Gears could be more properly meshed and alignment is problematic.	
Object Manipulation and Degrees of Freedom	Object manipulator design is excellent. Robot has 2 or more degrees of freedom.	Object manipulator design is very good in general. Some elements of the design could be more conducive to the component purpose. Robot has more than 1 degree of freedom.	Object manipulator design is good. Many elements of the design could be more conducive to the component purpose. Robot has only 1 degree of freedom.	Object manipulator design needs improvement. Most elements of the design could be more conducive to the component purpose.	
Sensors and Programming	Program works flawlessly and is easy to understand and modify. Robot uses 3 or more autonomous sensor functions.	Program works but is hard to understand or modify. Robot uses 3 autonomous sensor functions.	Program and one or more autonomous sensors do not work properly. Much more attention to the programming is required.	Program and autonomous sensors do not work.	
Overall	Robot design and programming are elegant. All components function smoothly, work well together, look like they belong together, and are easy to modify or replace. Efficient sensors and parts use.	Robot lacks some critical design components. Most components function smoothly, work well together, look like they belong together, and are easy to modify or replace. Efficient sensors and parts use.	Robot lacks many critical design components. Some components function smoothly, work well together, look like they belong together, and are easy to modify or replace. Parts and sensors use could have been more efficient.	Robot lacks most critical design components. Few components function smoothly, work well together, look like they belong together, and are easy to modify or replace. Parts and sensor use could have been more efficient.	

Total: / 20

Robotics 801A
COMPETITION: PART A
ASSESSMENT RUBRIC: ROBOT PERFORMANCE

Categories	10 Marks	9 marks	8 Marks	5 Marks	3 Marks	0 Marks
Competition	Robot completed competition task more efficiently than all class competitors. (1st Place)	Robot completed competition task more efficiently than all class competitors except 1. (2nd Place)	Robot completed competition task more efficiently than all class competitors except 2. (3nd Place)	Robot completed assigned task.	Robot completed part of task.	Robot did not complete any part of the assigned task OR Robot was not completed in time for competition.

Total: / 10

Robotics 801A
COMPETITION: PART B
ASSESSMENT RUBRIC: ROBOT PERFORMANCE

Categories	10 Marks	9 marks	8 Marks	5 Marks	3 Marks	0 Marks
Competition	Robot completed competition task more efficiently than all class competitors. (1st Place)	Robot completed competition task more efficiently than all class competitors except 1. (2nd Place)	Robot completed competition task more efficiently than all class competitors except 2. (3nd Place)	Robot completed assigned task.	Robot completed part of task.	Robot did not complete any part of the assigned task OR Robot was not completed in time for competition.

Total: / 10

Appendix B

STSE Research Project

Project: Science, Technology, Society, and the Environment
Option A

Task

Select an existing robotics technology of interest and identify its relationship among science, technology, society, and the environment. Add other pertinent information as called for in the criteria below.

You are permitted to work individually or with a partner. Prepare an electronic multimedia presentation to assist in communicating the results of your project.

STSE Presentation Criteria

Use the following questions/instructions to guide your research and presentation:

- Describe your robotics technology.
- Identify the uses of your robot.
- Describe the history and evolution of your robotics technology.
- Create a flowchart that outlines the programming of the robot's functions.
- What are 3 advantages of your robot?
- What are 3 disadvantages of your robot?
- Identify various careers associated with the production and maintenance of your robot.

STSE Presentation Logistics and Assessment

- Presentation to include all criteria listed above, graphics, and references
- Presentation duration to be 10-20 minutes, including questions
- Presentation to be assessed in accordance with the attached oral presentation rubric

Examples of Existing Robotics Technologies

DARPA Grand Challenge Driverless Car, Predator Unmanned Aerial Vehicle, Canada Arm, Dextre, NASA Pathfinder, NASA Sojourner, Honda ASIMO, Robotic surgery, Roomba vacuum cleaner, Husqvarna automower, Industrial robots, etc.

Assessment/Evaluation

Presentation	40%	(delivery and content of slideshow - see presentation rubric)
Notes	20%	(notes taken on presentations)
Test	40%	(students will be allowed to use their notes)

Project: Science, Technology, Society, and the Environment
Option B

Task

Identify a need for a radio-controlled or autonomous robot to help a particular group in society. Add other pertinent information as called for in the criteria below.

You are permitted to work individually or with a partner. Prepare an electronic multimedia presentation to assist in communicating the results of your project.

STSE Presentation Criteria

Use the following instructions to guide your research and presentation:

- Describe your robotic device and explain its function. Include a sketch.
- Identify the demographic that your robot is intended to serve.
- Identify the need for your robot's use.
- Identify similar technologies that are presently available and their cost.
- Estimate what it would cost to build your robot.
- Create a flowchart that outlines the programming of the robot's functions.
- Highlight what is different about your design (3 pros and 3 cons) with respect to what is presently available in society.

STSE Presentation Logistics and Assessment

- Presentation to include all criteria listed above, graphics, and references
- Presentation duration to be 10-20 minutes, including questions
- Presentation to be assessed in accordance with the attached oral presentation rubric

Examples of Societal Needs for Robotics Technologies

A robotic arm that will attach to a wheelchair to assist an individual reach items stored high on a shelf or open a door, a robot created to clean floors, a robot designed to help invalids in and out of bed, a robot created to shovel snow, etc.

Assessment/Evaluation

Presentation	40%	(delivery and content of slideshow - see presentation rubric)
Notes	20%	(notes taken on presentations)
Test	40%	(students will be allowed to use their notes)

Robotics 801A – STSE PRESENTATIONS - OPTION A
Presentation Rubric

Presentation:	
Group Members:	

Category	4 Marks	3 Marks	2 Marks	1 Mark	
Delivery	Speaker's voice is strong and clear. Speaker conveys confidence about the topic. Excellent eye contact.	Good speaking voice. Speaker is in command of topic. Good eye contact.	Clarity of speech is uneven. Speaker is not completely sure of topic. Limited or sporadic eye contact.	Control of speaking tone, clarity, and volume is not evident. Speaker does not convey interest. No eye contact.	
Description and Use	Excellent explanation and description of the robot's function and use. Subject knowledge is excellent	Clearly identifies the robot's description and use. Subject knowledge appears to be good.	Somewhat describes and identifies the robot's use, but there are 1-2 factual errors.	It is unclear what the robot does. Content is minimal and/or there are several factual errors.	
History and Evolution	Presents the complete history and evolution of the robot's design. In-depth description with details and graphics.	Includes essential knowledge about the robot's history and evolution, with some supporting details and graphics.	Somewhat explains the robot's history and evolution, but there are 1-2 factual errors.	Does not explain when the robot was invented or how it changed over time.	
Programming Flowchart	The programming flowchart clearly outlines exactly how the robot will function and complete its task. No steps are missing.	The programming flowchart clearly outlines what the robot is supposed to do, but one or two steps are missing.	The programming flowchart somewhat outlines what the robot is supposed to do, but several steps are missing.	Programming flowchart not provided.	
Advantages	Highlights 3 advantages of the robot's use.	Highlights 2 advantages of the robot's use.	Highlights 1 advantage of the robot's use.	Does not highlight any advantages.	
Disadvantages	Highlights 3 disadvantages of the robot's use.	Highlights 2 disadvantages of the robot's use.	Highlights 1 disadvantage of the robot's use.	Does not highlight any disadvantages.	
Use of Graphics	All graphics are attractive (size and colours) and support the theme/content of the presentation.	A few graphics are not attractive, but all support the theme/content of the presentation.	All graphics are attractive but a few do not support the theme/content of the presentation.	Several graphics are unattractive and detract from the content of the presentation.	
Sources		All text and graphics used in the presentation are properly referenced.	Some text and graphics used in the presentation are properly referenced.	None of the text or graphics used in the presentation are properly referenced.	
				TOTAL	

Robotics 801A – STSE PRESENTATIONS - OPTION B
Presentation Rubric

Presentation:	
Group Members:	

Category	4 Marks	3 Marks	2 Marks	1 Mark	
Delivery	Speaker's voice is strong and clear. Speaker conveys confidence about the topic. Excellent eye contact.	Good speaking voice. Speaker is in command of topic. Good eye contact.	Clarity of speech is uneven. Speaker is not completely sure of topic. Limited or sporadic eye contact.	Control of speaking tone, clarity, and volume is not evident. Speaker does not convey interest. No eye contact.	
Description	Description shows a large amount of original thought. Ideas are creative and inventive.	Description shows some original thought. Work shows new ideas and insights	Uses other people's ideas (giving them credit), but there is little evidence of original thinking.	Uses other people's ideas, but does not give them credit.	
Demographic, Need and Cost	Excellent explanation of what demographic or need the robot will meet. A detailed estimated cost is presented.	Clearly identifies what demographic or need the robot will meet. The estimated cost seems reasonable but few details are provided.	Somewhat identifies what demographic or need the robot will meet. The estimated cost seems unreasonable and no details are provided.	It is unclear what group or need this design will meet. Estimated cost is not provided.	
Presently Available Solutions and Cost	Covers available technology in-depth with details and cost examples. Subject knowledge is excellent	Includes essential knowledge about available technology and costs. Subject knowledge appears to be good.	Includes essential information about available technology but there are 1-2 factual errors.	Content is minimal and/or there are several factual errors.	
Programming Flowchart	The programming flowchart clearly outlines exactly how the robot will function and complete its task. No steps are missing.	The programming flowchart clearly outlines what the robot is supposed to do, but one or two steps are missing.	The programming flowchart somewhat outlines what the robot is supposed to do, but several steps are missing.	Programming flowchart not provided.	
Proposed Design Advantages and Disadvantages	Highlights 3 pros and 3 cons of the proposed design with respect to what is currently available in society.	Highlights 2 pros and 2 cons of the proposed design with respect to what is currently available in society.	Highlights 1 pro and 1 con of the proposed design with respect to what is currently available in society.	Does not highlight how the proposed design is better or worse than what is currently available in society.	
Use of Graphics	All graphics are attractive (size and colours) and support the theme/content of the presentation.	A few graphics are not attractive, but all support the theme/content of the presentation.	All graphics are attractive but a few do not support the theme/content of the presentation.	Several graphics are unattractive and detract from the content of the presentation.	
Sources		All text and graphics used in the presentation are properly referenced.	Some text and graphics used in the presentation are properly referenced.	None of the text or graphics used in the presentation are properly referenced.	
				TOTAL	

Robotics 801A – STSE PRESENTATIONS - OPTION A
Report Form

Existing Robotic Technology	______________________________ ______________________________
Team Members	______________________________
Description	
Uses	
History & Evolution	
Programing Flowchart	
Pros/ Advantages	
Cons/ Disadvantages	
Careers	

Robotics 801A – STSE PRESENTATIONS - OPTION B
Report Form

Proposed Robotic Technology	______________________________ ______________________________
Team Members	______________________________
Description	
Demographic or Need	
Presently Available Solutions and Cost	
Cost of New Design	
Programing Flowchart	
Pros/ Advantages	
Cons/ Disadvantages	

Appendix C

Skills Logbook

Technical Skills Rubric

~ Rating Scale ~

The following rubric will be used to describe students' progress as they develop and refine their technical skills related to the course.

Attempted	Beginning	Developing	Accomplished
0	***1***	***2***	***3***
Has not completed the required outcome.	**Completed the required outcome with supervision.** **Able to complete the task but requires instructor intervention.**	**Completed the required outcome with assistance.** **Demonstrates the ability to complete the task by using a guided plan and/or peer assistance to help solve problems that arise.** **Instructor acts only as a facilitator.**	**Completed the required outcome independently.** **Demonstrates the ability to complete the task and problem solve in a self-directed manner.** **Identifies and locates resources** **Identifies, selects, and demonstrates the use of tools, programing code, and materials efficiently and effectively.**

Robotics 801A

Student Name:_______________
Instructor:_______________________
School:_______________

Logbook - Learning Outcomes

801A Robotics	**Task Development Log**					**Date:**			
	Dates I worked on outcome					**Self** Rating	Initials	**Teacher** Rating	Initials
1. Safety Students will be able to practice workplace safety at all times and be able to demonstrate the proper and safe use of hand tools.									
2. Drivetrain and Gears Students will be able to understand and build robots that have different drivetrain and gear systems.									
3. Center of Gravity and Degrees of Freedom Students will be able to understand and build robots that take into account center of gravity concerns and different degrees of freedom.									
4. Sensors Students will be able to understand the function and use of seven different programmable sensors.									
5. Computer Programming Students will be able to use easyC Pro to program autonomous sensor functions for their robot.									
6. Engineering Design Process Students will be able to work collaboratively to carry out and document an engineering design process to meet predetermined challenges.									

Attendance: Days Absent — Teacher

Sept	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Total:	

Competence Rating: 0- Not Completed 1- Beginning 2- Developing 3- Accomplished

Robotics 801A

Student Name:_______________

Instructor:______________________

School:_______________

Logbook - Achievement Indicators

801A Robotics		Date:			Date:		
#	**Task**	Self	Teacher Rating	Initials	Self	Teacher Rating	Initials
1.1	Uses hand tools and shop equipment						
1.2	Maintains safe work environment						
2.1	Configure a transmitter and micro controller effectively and accurately to control a robot						
2.2	Understands simple and complex gear systems						
3.1	Explains how center of gravity impacts robot performance						
3.2	Applies different degrees of freedom to robot manipulators						
4.1	Uses bumper and limit switches to accomplish a task						
4.2	Uses light sensor to accomplish a task						
4.3	Uses ultrasonic sensor to accomplish a task						
4.4	Uses optical shaft encoder to accomplish a task						
4.5	Uses potentiometer to accomplish a task						
5.1	Uses flowcharts and syntax to break a complicated behaviour into a series of simple tasks						
5.2	Uses programming code to accomplish a task						
5.3	Diagnoses and troubleshoots programming code						
6.1	Defines problems/challenges to be solved						
6.2	Analyses and evaluates possible solutions						
6.3	Designs and constructs a robot with autonomous and RC functions to solve different kinds of nonfamiliar problems in both conventional and innovative ways						
6.4	Documents the design, testing, programming, and construction of a robot						
6.5	Communicates effectively with others						
6.6	Demonstrates ability to work effectively and respectfully with each team member						

Competence Rating: 0- Not Completed 1- Beginning 2- Developing 3- Accomplished

ROB801A
Outcome (s): ____________

Student Name:____________
Date: ____________

Evidence of Learning

What I Know (prior knowledge and experiences)

What I Need to Know (concerns, questions, skills, procedures)

Learning Plan (opportunities to extend learnings *ex. courses, ASAP, Community-Based Learning*)

Course: ________________________ **Student Name:**________________
Date: ________________

Learning Journal

Related Notes / Drawings / Comments

Logbook & Learning Journal Assessment Rubric

Category	4 Marks	3 Marks	2 Marks	1 Mark
Logbook Entries	Self-assessments in the Logbook are complete and demonstrate critical reflection on learning experiences and processes.	Self-assessments in the Logbook are complete and reflect on learning experiences and processes.	Self-assessments in the Logbook are incomplete or do not accurately reflect learning experiences and processes.	Self-assessments in the Logbook are incomplete and do not accurately reflect learning experiences and processes.
Evidence of Learning	Makes statements that are accurate and well supported with evidence (e.g., Logbook).	Usually, but not always, makes statements that are accurate and well supported with evidence (e.g., Logbook).	Makes several inaccurate statements and/or supports few statements with evidence (e.g., Logbook).	Consistently makes inaccurate statements and/or fails to provide statements with evidence (e.g., Logbook).
Learning Plan	Develops a Learning Plan related to the learning, personal strengths and weakness. Shows a keen interest in the class and/or the skilled trade.	Develops a Learning Plan related to personal strengths and weakness. Committed to the class and/or the skilled trade.	Learning Plan in not clearly defined with a minimal connection between the experience and the learning. Unchallenged. Somewhat committed to class and/or the skilled trade.	No clear Learning Plan with no clear connection between the experience and the learning. Not committed to the class or the skilled trade.
Learning Journal	Clearly expresses technical concepts using notes, descriptions, illustrations, and technical information.	Usually, but not always, expresses technical concepts using notes, descriptions, illustrations, and technical information.	Uses brief or general statements with few details to explain technical concepts.	Considerable difficulty clearly expressing technical concepts, uses brief statements glossing over event(s).
Submitted on Time		Submits Logbook and Learning Journal on time.	Difficulty meeting deadlines related to the Logbook and Learning Journal	Consistently fails to submit Logbook or Learning Journal.
Writing Quality			Makes few or no typographical, spelling, and/or grammatical errors.	Consistently makes typographical, spelling, and/or grammatical errors.

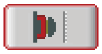
Total: /21

Appendix D

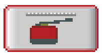
easyC Pro Help Topics

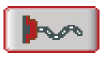
Legend of easyC Pro Block Icons

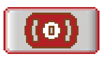
Inputs:

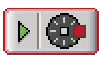
 Bumper switch Sensor

 Light Sensor

 Limit Switch Sensor

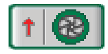
 Line Following Sensor

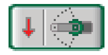
 Potentiometer Sensor

 Optical Shaft Encoder Sensor

 Ultrasonic Sensor

Outputs:

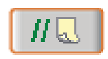
 Motor Module

 Servo Module

Program Flow:

 Wait

 Print to Screen

 Comment

User Code

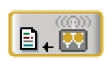
RC Control:

2 Motor Arcade

4 Motor Arcade

2 Motor Tank

4 Motor Tank

 Rx Input

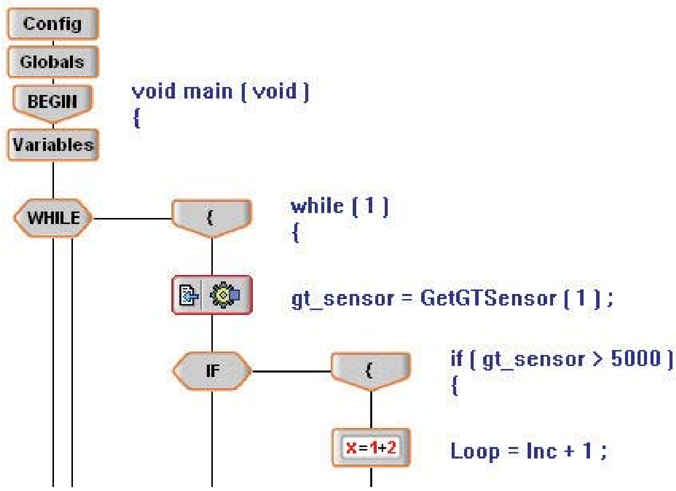
Commonly Used Operators:

Operator	Definition	Example syntax	Example meaning
<	less than	x<0	value "x" is less than zero
<=	less than or equal to	x<=0	value "x" is less than or equal to zero
>	greater than	x>0	value "x" is greater than zero
>=	greater than or equal to	x>=0	value "x" is greater than or equal to zero
==	equal to	x==0	value "x" is equal to zero
!=	not equal to	x!=0	value "x" is not equal to zero
||	or	x>0 || y<0	value "x" is greater than zero OR value "y" is less than zero
&&	and	x>0 && y<0	value "x" is greater than zero AND value "y" is less than zero

Fundamentals

easyC Pro ♦ Fundamentals ♦ Block Programming Window

The Block Programming Window is where you create the flowchart version of your programs. Every new block program starts out with a Configuration block, a Global Variables block, a Begin block, a Local Variable definition block and an End block. The Robot Configuration should be defined before any other blocks are added to the program. This information is necessary so that you can properly assign inputs and outputs, and to help you as you write your program.



Function blocks are the primary means of creating programs in easyC. Each block usually represents a call to a single function, or a single line of code. A function block is used by dragging it from the function tree into the block programming window. As you drop blocks, the equivalent C code is automatically generated in the C-code Window.

You can only insert blocks into the flow chart in a location that results in logical, syntax-correct code. If you attempt to insert a block into an illegal area the cursor icon changes to display and the program will not allow the block to be added. See a video of this in action .

Modifying the Block Program

To modify your program directly from the flow chart:

- To move a block click and drag the block to a new section of the flow chart
- Right-click on any of the blocks to *bring up a list of available editing tools* , such as Delete, Edit, Cut, Copy and Paste.

The block diagram display can be scaled. To increase or decrease the size of the icons in the flow chart, click the Zoom in or Zoom out icon in the toolbar.

Keyboard Shortcuts

There are several keyboard shortcuts available to the user to speed up productivity in the Block Programming Window.

- Control Select: By holding down the control key and dragging a block, you will create a duplicate copy of that block.
- Shift Select: By holding down the shift key, you can select multiple continuous portions of code. All the code you select using multiple select must contained in the same vertical level in the Block Programming Window. You can copy, paste, and delete multiple block selections.
- Cut, Copy: Make your selection in the block window and use the Ctrl + X, Ctrl + V keys as a shortcut.

You can change the way the flow chart appears using the Flow Chart setup dialog box.

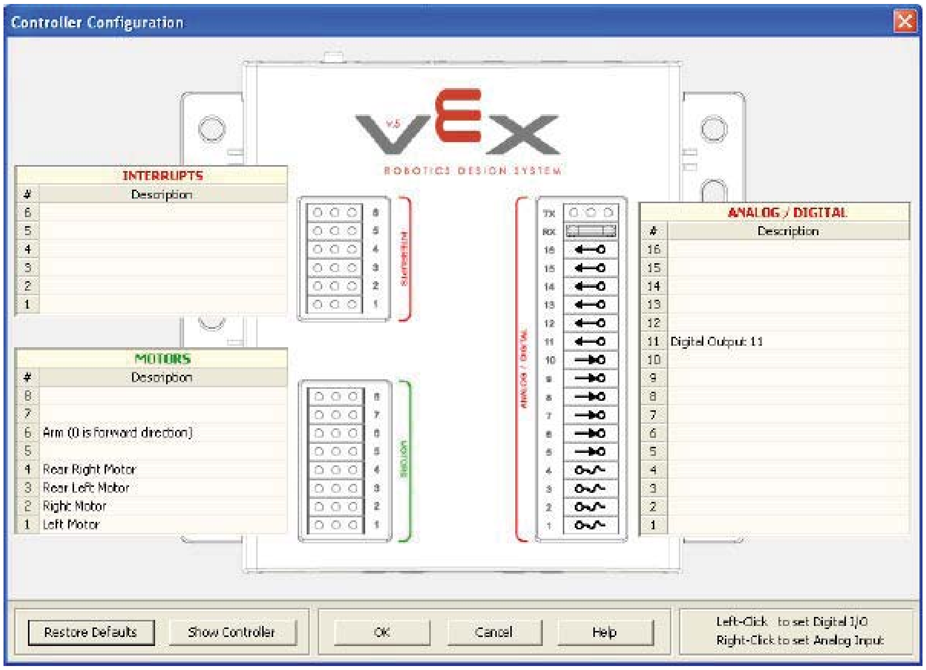
See also:

- Config Block
- Globals Block
- Begin Block
- Variables Block

easyC Pro ♦ Fundamentals ♦ Robot Configuration

The Controller Configuration page is where you define your robot map. Here you can label each of the motors and sensors connected to your robot, and include notes about the directions and characteristics of each. These descriptions appear in the Online Window and in the description field of each dialog box. You can also edit the descriptions directly from the Online Window.

You can print your robot map by selecting 'Print Controller Configuration' from the File Menu.



The Digital IN/OUT Bank on the Vex controller consists of 16 ports that can be configured to work as Digital Inputs, Digital Output, or Analog Input. The ports must be configured to match the type of sensor that is plugged into them (digital input for a digital sensor, etc.). The configuration you choose here will be mirrored in the Online Window, and will determine the ports available for selection in each sensor dialog box.

- Left click on an I/O port to define it as a digital input (arrow pointing towards the pin), a digital output (arrow pointing away from the pin), or analog input (squiggly line).
- Click on Show Controller *to see a picture* of the Robot Controller.

easyC Pro ♦ Fundamentals ♦ Defining Variables

A variable is a value-holding "container" with a name (label). Any time you need to store a value for later use in a program, use a variable to store that bit of information. You can then retrieve the value when it is needed. Variables, unlike Constants, can be overwritten any number of times through the course of a program. There are two types of variables in easyC, Local Variables and Global Variables.

Variables can be defined in three locations.

1. Global Variables are defined using the Program Globals dialog (F6).

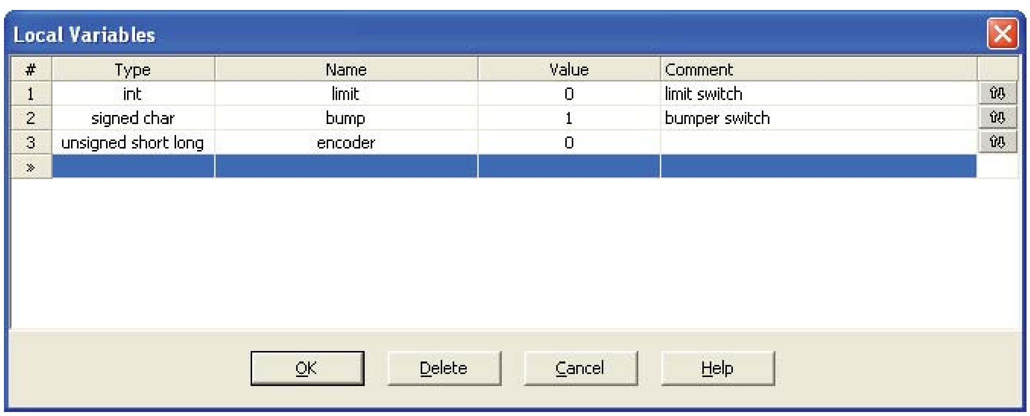

2. Local Variables can be defined as part of a user function definition.


3. Local Variables are defined in the Local Variables dialog box (Ctrl+F6).


Note:
Variables used but not defined in any of these three locations will cause compilation errors.

Although they may serve different purposes, variables in easyC are always defined in a similar manner.

- To define and edit variables, open the appropriate dialog box. The Local Variables dialog box is provided as an example below.



- In the Type field, use the drop down list to select from all the available variable types. View definitions of all the variable types available in easyC
- You must name each new variable:
  - Use any uppercase or lowercase letter, number or underscore.
  - Do not use any other character or spaces
  - The first character cannot be a number

  - The variable name cannot consist of only numbers
- You can set the variable to an initial value and write comments to describe the variable. Variable comments will appear in the description fields of appropriate easyC dialogs.
- For Local Variables, you can also the value to be another variable, so long as the variable value is already defined.
- You can change the order of the variables in the dialog box, by clicking on the double arrow symbol, and dragging your mouse either up or down.
- Use the delete button to delete the currently selected variable.
- As a general rule, you cannot close a variable dialog box until all the necessary fields have been satisfactorily completed.
- You can also define and work with Arrays.

easyC Pro ♦ Fundamentals ♦ Global Variables

A variable is a value-holding "container" with a name (label). Any time you need to store a value for later use in a program, use a variable to store that bit of information. You can then retrieve the value when it is needed. There are two types of variables in easyC, Local Variables and Global Variables.

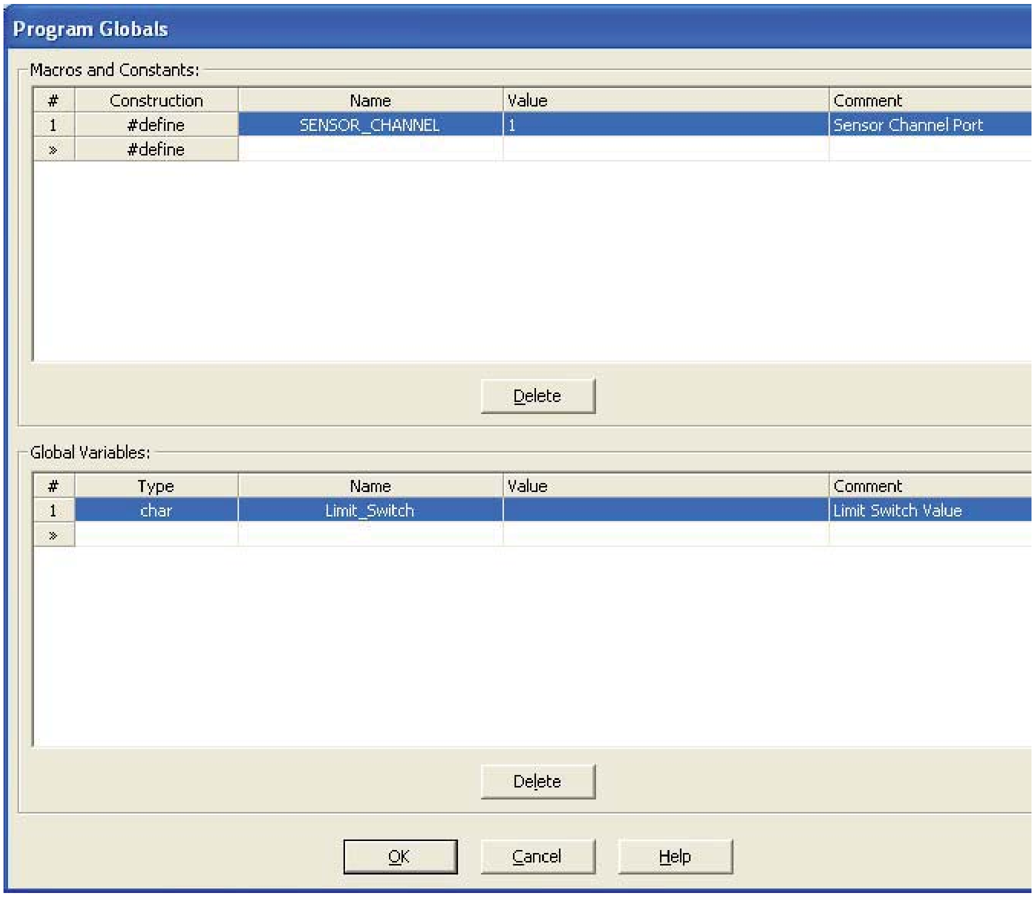
A Global Variable, sometimes referred to as an external variable, is one which can be seen by every function in your project. To a programmer, a Global variable has indefinite scope because it is always visible. As a consequence, a Global Variable is also modifiable everywhere within your project, and will always have the same value everywhere. If you create a user function, and assign a Global Variable to a new value within that function (the integer 2 for example), that Global Variable will have the value 2 everywhere in your project, not just in your new user function.

Accessing Program Globals

The Program Globals dialog box is accessible in four ways.

- By double clicking on the 'Globals' block in the 'main' function of your project.
- On the Main Toolbar, by finding 'Program Globals' in the Options menu.
- By accessing the dialog directly on the main toolbar, by clicking on the icon 
- By pressing the 'F6' key on most keyboards.

Understanding the Program Globals dialog box



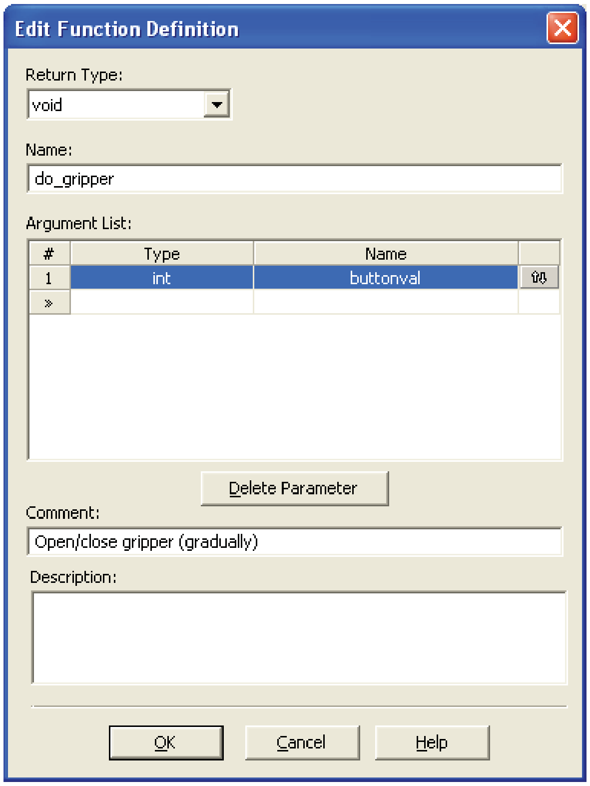
- Use the Macros and Constants section to define your Global Macros and Constants.
- Use the Global Variables section to define your Global Variables.
- Enter a value to specify an initial value for your variable. This value will be replaced when the program assigns a new value to the variable.
- Enter a comment which will appear in the description field of many dialog boxes. A comment can be helpful in remembering the purpose of each variable. Variable comments will appear in the description fields of appropriate easyC dialogs.

See also: Defining Variables

easyC Pro ♦ Fundamentals ♦ Begin Block

BEGIN

At the beginning of every function in easyC is the Begin Block. In the Main Function you are only allowed to add comments using the Begin Block, because the Main function does not call or retrieve values. In your user defined functions, you can use the Begin Block to edit your function definitions, much as you would when creating a new function for the first time.



- The Return Type should be set to the variable type your function is returning. If your user function does not return a value, leave this field void.
- In the *Name* field, enter a name for your new function.
- In the Argument List area, you can assign any values your function is calling to variables. Use the drop down list to select a type, and enter a name by selecting a predefined variable, or enter your own. This process is similar to how you would define a variable.

Note:
This is not the correct place to define general variables for your new user function. Variables defined here are associated respectively with values being initialized with your user function. The Argument List is order specific, with the first variable referring to the first called value.

- You can change the order of the variables in the dialog box, by clicking on the double arrow symbol, and dragging your mouse either up or down.

- In the Code area, view a preview of the C code as its generated.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block. Comments placed here will only appear when working in the user function, and not when the function is called.
- In the Description field, enter a description for this function. Your function description will not be visible in the block programming window, and is only referenced when importing and exporting functions (ex. when using the Add Existing Function Option)

Note:
Making any changes in the Begin block that result in any variables changing type, or in variables changing type with respect to associated arguments in the calling function, will result in the arguments in the function call being reset.

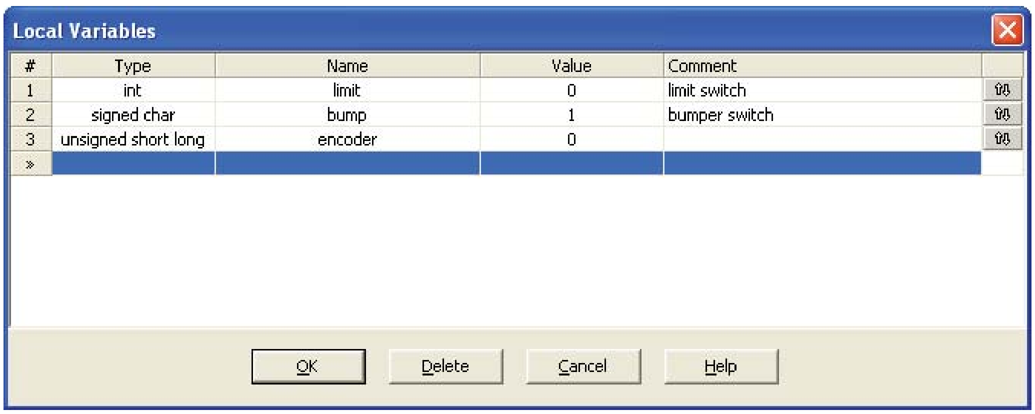
easyC Pro ♦ Fundamentals ♦ Local Variables

A variable is a value-holding "container" with a name (label). Any time you need to store a value for later use in a program, use a variable to store that bit of information. You can then retrieve the value when it is needed. There are two types of variables in easyC, Local Variables and Global Variables. Local values reinitialize each time the function they are contained within is called.

Accessing Local Variables

The Local Variables dialog box is accessible in four ways.

- By double clicking on the 'Variables' block on the flow chart in your User Function.
- On the Main Toolbar, by finding 'Program Local Variables' in the Options menu.
- By pressing the 'Ctrl + F6' keys on most keyboards.



Use the program variables dialog box to define all the variables used in the block program.

View definitions of all the variable types available in easyC

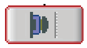
Once defined, the variables are local, meaning the definitions are stored only in the function they are defined in. Local variables are not visible or retrievable in any other function.

- In the Type field, use the drop down list to select from all the available variable types .
- You must name each new variable:
  - Use any uppercase or lowercase letter, number or underscore.
  - Do not use any other character or spaces
  - The first character cannot be a number
  - The variable name cannot consist of only numbers
- You can set the variable to an initial value and write comments to describe the variable. Variable comments will appear in the description fields of appropriate easyC dialogs.

See also: Defining Variables

Function Blocks: Inputs

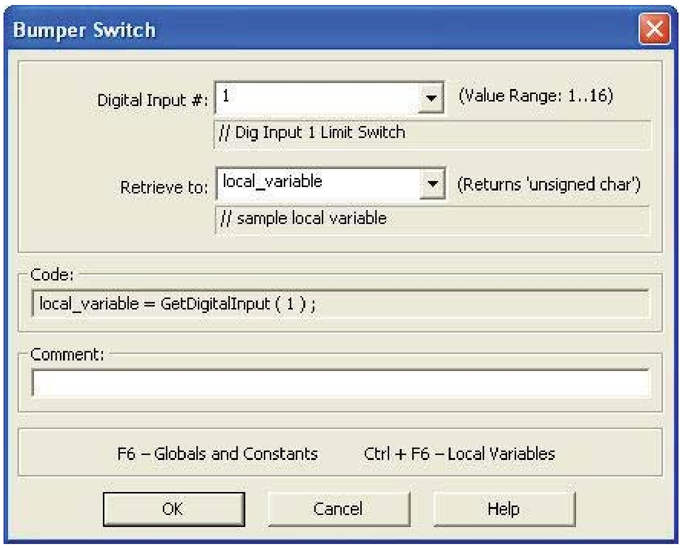
easyC Pro ♦ Function Blocks ♦ Bumper Switch



Use the Bumper Switch function block to define behavior from a simple on/off pressure switch, such as the one found in the Vex Starter Kit.

A Bumper Switch is a digital sensor . It can distinguish between two states: whether it is pressed or not pressed. When the switch is not pressed, the robot interprets this value as a 1 (one). When the switch is pressed the robot interprets the value as a 0 (zero).

When you drag a Bumper Switch block into the Programming Window, the Bumper Switch dialog box appears.



- Choose the Digital Input # that matches the digital input to which the switch is wired on the controller. Only ports defined as digital inputs in the controller configuration screen are shown.
- Select a predefined variable in the "Retrieve to" field. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling. You may hit either F6, or Ctrl + F6 at any time to open the variable definition windows.
- A description field lies below each combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

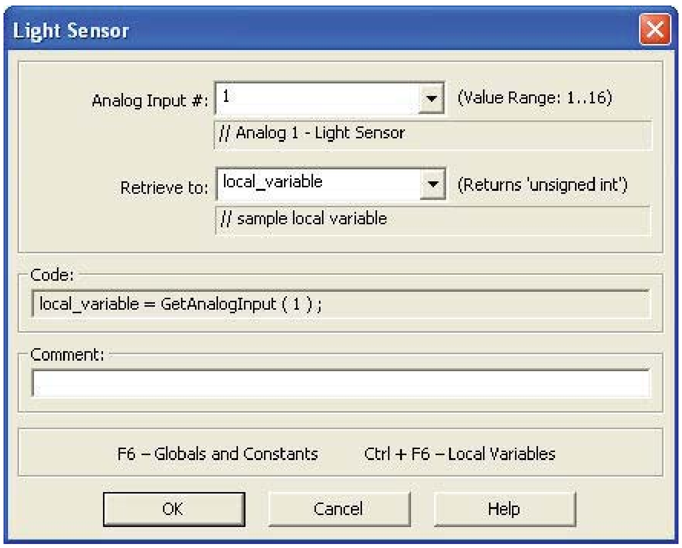
easyC Pro ♦ Function Blocks ♦ Light Sensor

The Light Sensor function block allows the user to define behavior based on signals from a Light Sensor, such as the one found in Vex.

The Light Sensor is an analog sensor . It can provide a range of feedback to the robot from 0 - 1024.

A Light Sensor can read the relative lightness of an area, translate the reading into a single numeric value, and send that value to the robot. A darker area receives a higher numeric value.

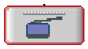
When you drag a Light Sensor block into the Programming Window, the Light Sensor dialog box appears.



- Choose the Analog Input # that matches the analog input to which the light sensor is wired on the controller. Only ports defined as analog inputs in the controller configuration screen are shown.
- Select a predefined variable in the "Retrieve to" field. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling. You may hit either F6, or Ctrl + F6 at any time to open the variable definition windows.
- A description field lies below each combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

- To see a sample program demonstrating how to program a Light Sensor, open the LIGHTTEST.ECP Project located in the Test Code folder.

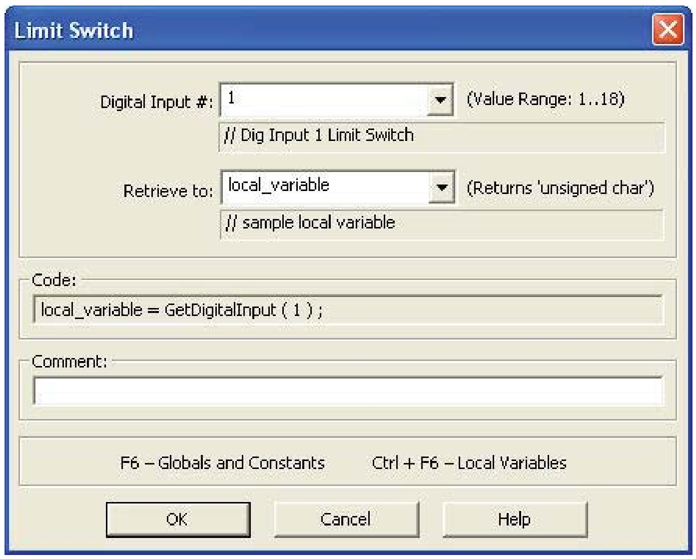
easyC Pro ♦ Function Blocks ♦ Limit Switch



The Limit Switch function block allows the user to define behavior based on signals from the Limit Switch.

The Limit Switch is a digital sensor . It can distinguish between two states: whether the limit switch is pressed or not pressed. When the Limit Switch is not pressed, the robot interprets this value as a 1 (one). When the Limit Switch is pressed the robot interprets the value as a 0 (zero).

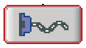
When you drag a Limit Switch block into the Programming Window, the Limit Switch dialog box appears.



- Choose the Digital Input # that matches the digital input to which the switch is wired on the controller. Only ports defined as digital inputs in the controller configuration screen are shown.
- Select a predefined variable in the "Retrieve to" field. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling. You may hit either F6, or Ctrl + F6 at any time to open the variable definition windows.
- A description field lies below each combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

- To see a sample program demonstrating how a Limit Switch is used, open the LIMITTEST.ECP Project located in the Test Code folder.

easyC Pro ♦ Function Blocks ♦ Line Follower

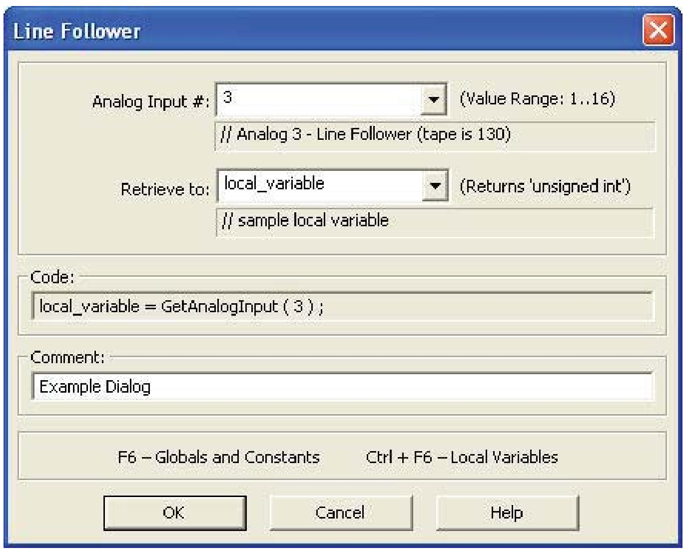


The Line Follower function block allows the user to define behavior based on analog signals from a Line Follower.

The Line Follower is an optical analog sensor . It can provide a range of feedback to the robot from 0 - 1024.

A Line Follower sensor illuminates a surface with infrared light and reads the amount of reflected light. The sensor translates the reading into a single numeric value, and sends that value to the robot. A darker area receives a higher numeric value.

When you drag a Line Follower block into the Programming Window, the Line Follower dialog box appears.



- Choose the Analog Input # that matches the analog input to which the Line Follower is wired on the controller. Only ports defined as analog inputs in the controller configuration screen are shown.
- Select a predefined variable in the "Retrieve to" field. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling. You may hit either F6, or Ctrl + F6 at any time to open the variable definition windows.
- A description field lies below each combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

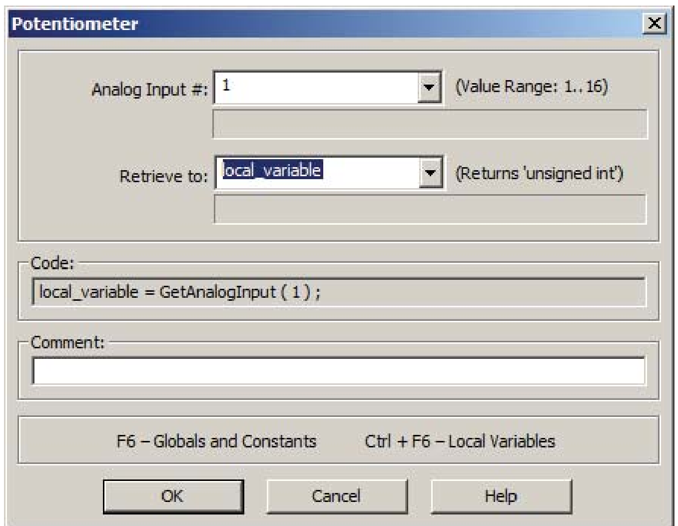
- To see a sample program demonstrating how a line follower is used, open the LINEFOLLOWER.ECP Project located in the Test Code folder.

easyC Pro ♦ Function Blocks ♦Potentiometer

The Potentiometer function block allows the user to define behavior based on analog signals from the Potentiometer sensor. The Potentiometer sensor is an analog sensor. It gets an analog measurement of the angular position. It describes an electrical device in which the user can adjust the resistance. As the resistance of the sensor changes, a varying voltage is created and thus the sensor acts as a variable voltage divider. This varying analog voltage can be measured by the Vex Controller and is proportional to the position of the shaft connected to the center of the Pot.

- To see a test program demonstrating how the Potentiometer is used, open the sample Project file (Potentiometer.ecp) located in the Test Code folder.

When you drag a Potentiometer block into the program window the Potentiometer Sensor dialog box appears.

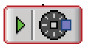


- choose the Analog Input # that matches the analog input to which the line follower sensor is wired on the controller. The Analog Input number can also be set to a variable for advanced applications.

  Note:
  Be sure the corresponding input port is defined as an analog input in the controller configuration.

- Select a predefined variable in the "Retrieve to" field. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling. This allows you to store the feedback provided by the Line Follower into a variable and is necessary to correctly define the block.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

easyC Pro ♦ Function Blocks ♦ Optical Shaft Encoder

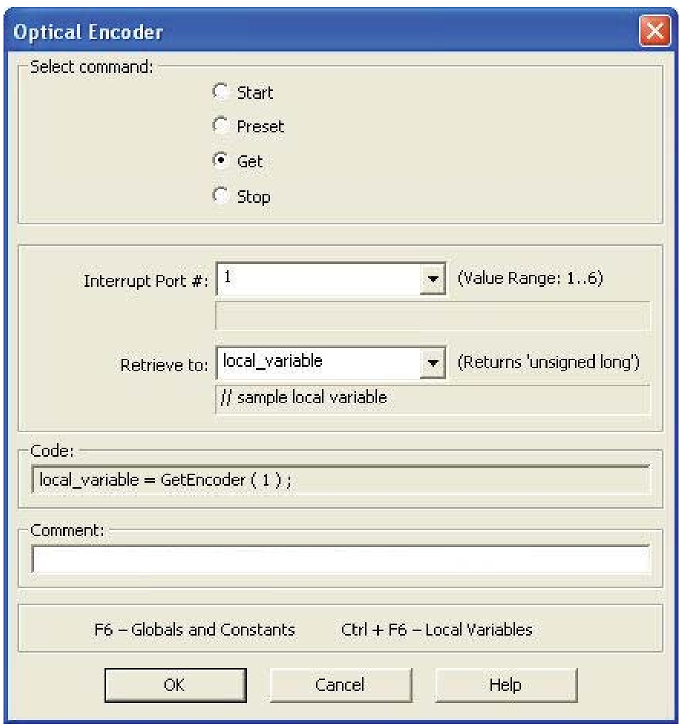


The Optical Encoder function block allows the user to define behavior based on signals from a simple (non quadrature) Optical Encoder, such as the one found in Vex.

An Encoder is a digital sensor whose signals can be used to measure and define rotational movement. The encoder uses a disk with equally-spaced slots around its outer edge. An infrared light sensor aimed toward the disk will generate a digital signal based on whether its light hits the disk or passes through one of the slots in the disk. When the encoder senses the disk, it generates a digital signal of 1; if it senses a slot, it generates a 0 (zero) signal.

As the disk rotates, the encoder generates a string of signals (i.e. 0101010). The encoder can use these signals to count and record the number of slots read, thereby reading the amount of rotation. A simple optical encoder can count in only one direction, and signals generated while running in reverse will be indistinguishable from the forward direction.

When you drag an Optical Encoder block into the Programming Window, the Optical Encoder dialog box appears, allowing you to choose one of four commands for the encoder function to process:



The ***Start** option* tells the encoders to start counting encoder pulses.

The ***Preset** option* allows the user to set the encoder to a fixed value or to the value of a variable.

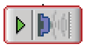
The ***Get** option* allows the user to store the feedback from the encoder into a variable (the default type to return is long).

The ***Stop** option* tells the encoders to stop counting encoder pulses.

- Select the Interrupt Port # that corresponds with the port the encoder is plugged into on the controller.
- Select a predefined variable in the "Retrieve to" field. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling. You may hit either F6, or Ctrl + F6 at any time to open the variable definition windows.
- A description field lies below each combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

- To see a sample program demonstrating how the Optical Shaft Encoder is used, open the ENCODERTEST.ECP Project located in the Test Code folder.

easyC Pro ♦ Function Blocks ♦ Ultrasonic Sensor

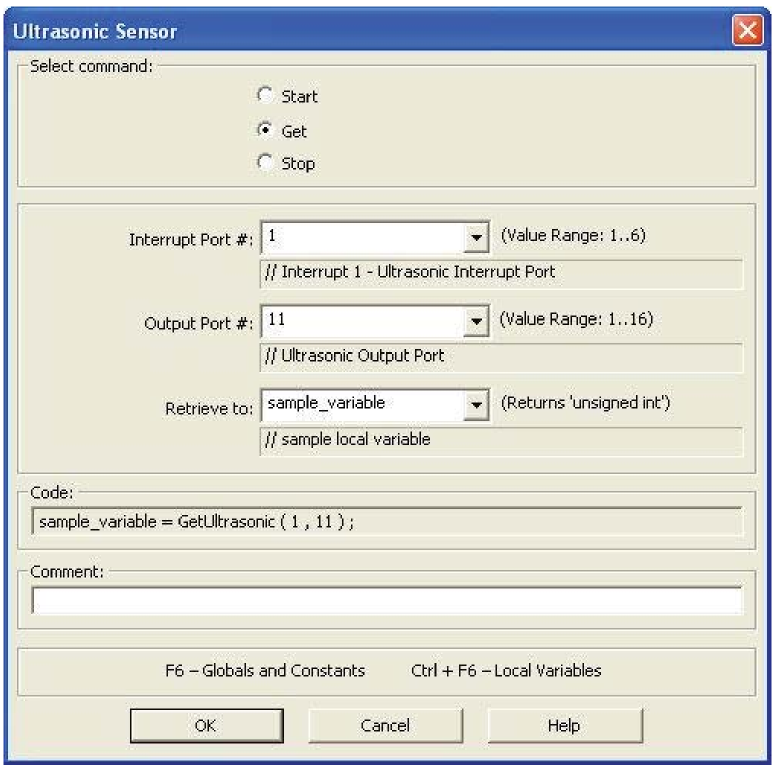


The Ultrasonic Sensor function block allows the user to define behavior based on analog signals from the Vex ultrasonic sensor.

The Ultrasonic Sensor uses high-frequency sound waves to detect objects. It emits a sound wave and measures how long it takes the sound wave to bounce back. The closer the detected object the lower the signal value. The measurement is translated into a numeric value from 2 - 100, reported in inches. If the sensor does not detect an object, it will read a high value, around 100.

- To see a sample program demonstrating how the Ultrasonic Sensor is used, open the ULTRASONICTEST.ECP Project located in the Test Code folder.

When you drag an Ultrasonic Sensor block into the Programming Window, the Ultrasonic Sensor dialog box appears.



The ***Start** command* tells the ultrasonic sensor to start recording sound waves.

The ***Get** command* allows the user to store the feedback from the sensor into a variable (the default type to return is unsigned int).

The ***Stop** command* tells the ultrasonic sensor to stop recording.

- Select the Interrupt Port # for the A channel that corresponds with the port on the Vex controller.

- Select the Input Port# for the B Channel that corresponds to the port on the Vex controller. Only ports defined as digital outputs in the controller configuration screen are shown.
- A description field lies below each combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

Note:
Vex Ultrasonic Sensors are labeled with reference to the sensor, so the input cable should be connected as a digital output on the robot controller.

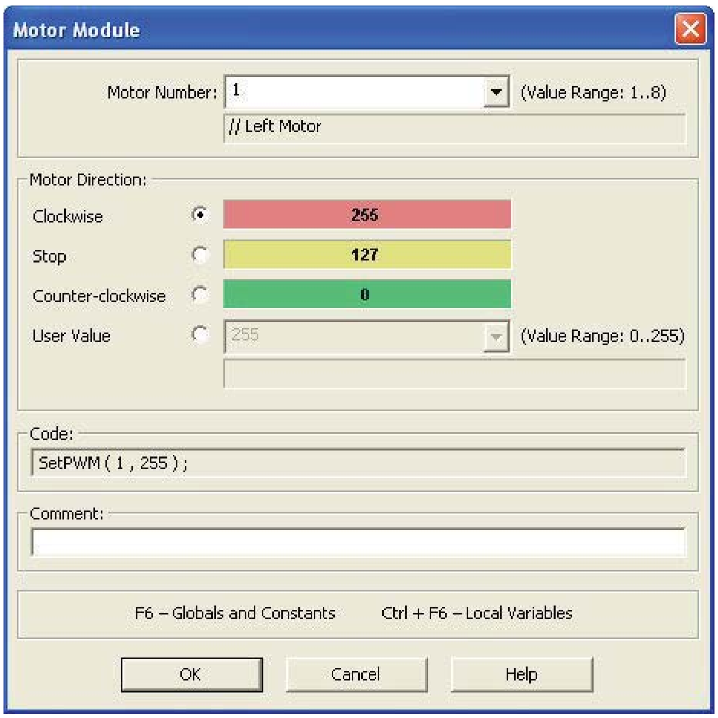
Function Block: Outputs

easyC Pro ♦ Function Blocks ♦ Motor Module

Use the Motor Module block to control the Motor Module.

The Motor Module can rotate continuously at a set PWM (pulse, width, modulation). The PWM value defines the direction and speed of rotation of the motor.

Note:
When the robot is turned on the motor module may jump slightly. This occurs as part of the controller's initialization procedure, even if you have not programmed any action for the motor.



- Select the PWM number that corresponds to the PWM port you wish to control. The PWM number can also be set to a variable.

Clockwise	
Set the PWM to 255, to be fully in the clockwise direction.	
Stop	
Set the PWM to 127, or off.	
Counter-clockwise	
Set the PWM to 0, to be fully in the counter-clockwise direction.	

User value

Sets the PWM to a fixed value or the value of a variable.
To set it to a variable value choose a predefined variable, or enter your own.

- A description field lies below each combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

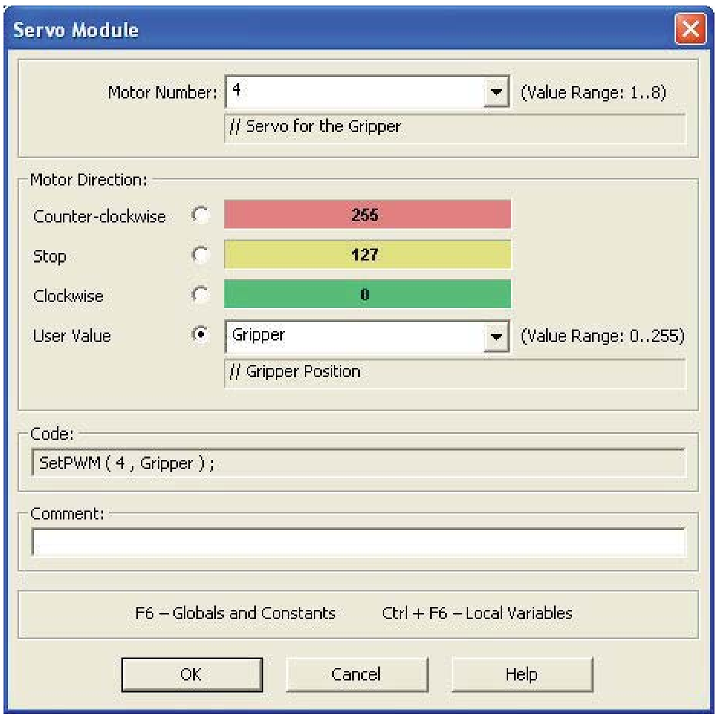
easyC Pro ♦ Function Blocks ♦ Servo Module

Use the Servo Module block to control a servo module.

A servo module has a limited range of motion (about 120 degrees). The PWM value specifies a position within that range of motion for the servo motor to move to and hold. Servos look very similar to motors, but will be labeled as a 'Servo Module' on the green facing.

Note:

When the robot is powered on, the servo motor defaults to a PWM value of 127, exactly in the middle of its range of motion. This occurs as part of the controller's initialization procedure, even if that motor port has not been programmed.



- Select the PWM number that corresponds to the PWM port you wish to control. The PWM number can also be set to a variable.

Clockwise

Set the PWM to 255, to be fully in the clockwise direction.

Stop

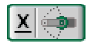
Set the PWM to 127, or off.

Counter-clockwise

Set the PWM to 0, to be fully in the counter-clockwise direction.

User value

Sets the PWM to a fixed value or the value of a variable.
To set it to a variable value choose a predefined variable, or enter your own.



- A description field lies below each combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

Function Blocks: Program Flow

easyC Pro ♦ Function Blocks ♦ If Condition

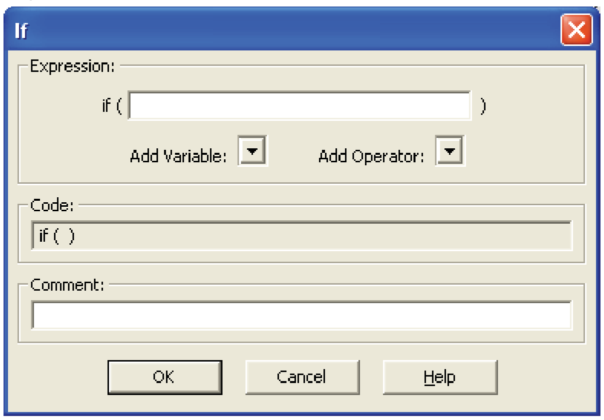
Use an If Condition to program behaviors that are dependent upon an expression you define.

When you have added the condition the program, braces appear in the window to signify the beginning and ending of the expression. This way the expression and programmed behaviors associated with the condition stand out from the rest of the program.

The If Condition will run a sequence of code as long as the specified expression is TRUE. If the expression is FALSE, it will skip the code within the brackets surrounding the condition and move to the next block of code.

Click to see a sample IF condition layout

When you add an If Condition to your program a dialog box opens allowing you to define the expression for the condition:



- Select a predefined variable using the Add Variable drop down list. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling.
- Use the Add Operator drop down list to select commonly used operators.

  See definitions of commonly used operators.

- You an also type the entire string directly into the dialog box.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

To keep braces from showing in the bock programming window

easyC Pro ♦ Function Blocks ♦ Else-If

Use an Else-If Condition as a clause to an already existing If Condition statement.

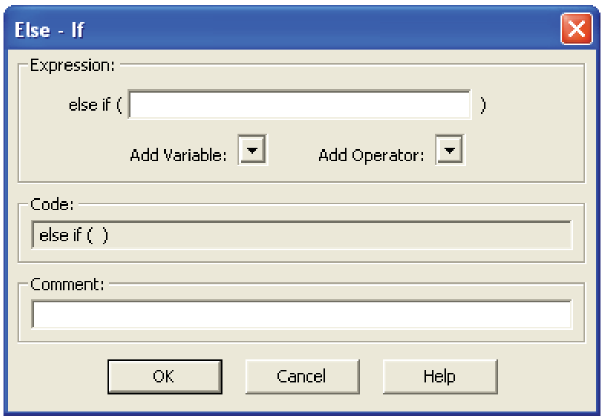
The Else-If Condition will run a sequence of code as long as the specified expression is TRUE. If the expression is FALSE, it will skip the code within the brackets and move to the next block of code. An Else-If Condition will only run if all of the preceding If and Else-If elements in the current If Condition structure were FALSE.

An Else-If Condition can only be placed so that it immediately follows an If Condition, or another Else-If Condition statement.

Click to see a sample Else-If Layout.

When you have added an Else-If Condition to your program, braces appear in the window to signify the beginning and ending of the condition. This way the expressions stand out from the rest of the program.

When you drag the Else-If command to your program a dialog box opens allowing you to define the expression:



- Select a predefined variable using the Add Variable drop down list. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling.
- Use the Add Operator drop down list to select commonly used operators.

  See definitions of commonly used operators.

- You can also type the entire string directly into the dialog box.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

To keep braces from showing in the bock programming window

Click to see the sample If/Else-If/Else condition provided with easyC

easyC Pro ♦ Function Blocks ♦ Else Condition

Use an Else Condition as a clause to an already existing If Condition statement.

The Else Condition will run a sequence of code whenever it is reached in an If Condition structure. The Else Condition must always be placed directly behind either an If or Else-If Condition, and may be preceded by many such conditions. Should all of the expressions in the If Structure prove FALSE, the Else Condition statement will always be TRUE, and the code within the brackets will be executed.

Click to see a sample Else Condition layout.

When you have added the Else Condition to your program, braces appear in the window to signify the beginning and ending of the condition. This way the expressions stand out from the rest of the program.

To keep braces from showing in the bock programming window

Click to see the sample IF/ELSE-If/Else condition provided with easyC

easyC Pro ♦ Function Blocks ♦ While Loop

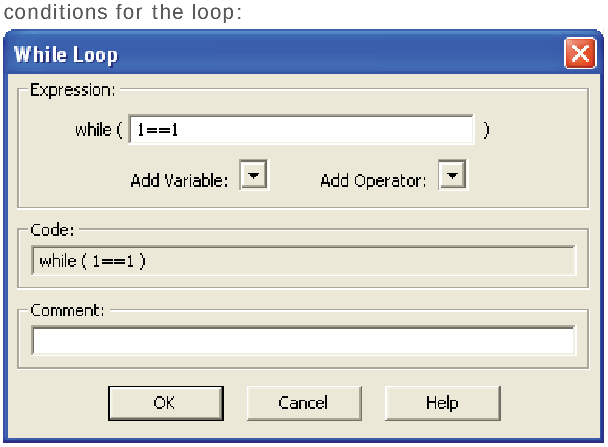
Use a While Loop to program behaviors that are dependent upon an expression you define.

The While Loop will repeat the code sequence for as long as the specified expression is TRUE. If the expression is FALSE, it will skip the code within the brackets surrounding the loop and move to the next block of code.

Click to see a sample WHILE loop layout

When you have added the loop the program, braces appear in the window to signify the beginning and ending of the loop. This way the expressions and programmed behaviors associated with the loop stand out from the rest of the program.

When you drag the while command to your program a dialog box opens allowing you to define the conditions for the loop:



- Select a predefined variable using the Add Variable drop down list. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling.
- Use the Add Operator drop down list to select commonly used operators.

  See definitions of commonly used operators.

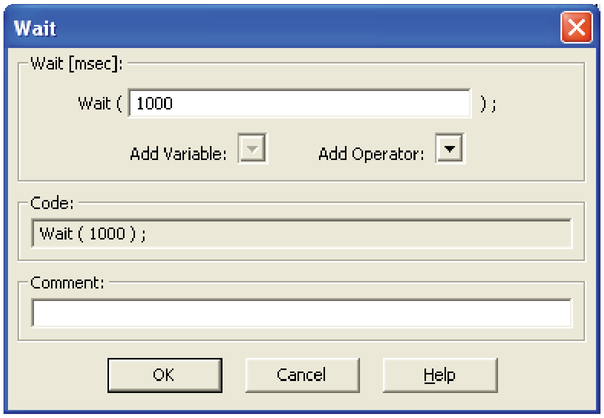
- You an also type the entire string directly into the dialog box.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

To keep braces from showing in the block programming window.

easyC Pro ♦ Function Blocks ♦ Wait

Use the Wait function block to pause the execution of the program for a specified period of time. Compare this with previous versions of the default code, which cannot wait without returning to the main loop. While your program is waiting, outputs will continue to function based on their last known information, and sensors will continue working in the background.

The time is specified in milliseconds. The actual number of milliseconds can be entered or the name of a predefined variable. If you enter a variable name, be sure to enter the name exactly as it is defined.



This function block is commonly used for dead reckoning sequences. For example, inserting a Wait function block between a Motor Module CW (255) command and a Motor Module OFF (127) command will specify how long the motor should run before it turns off.

View example

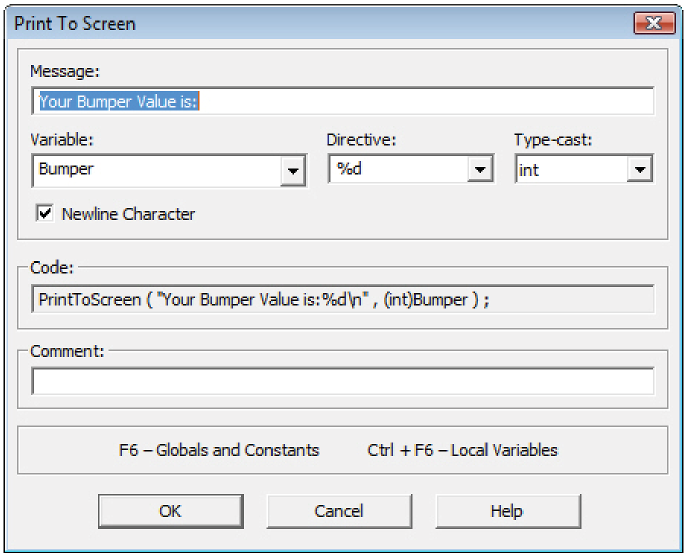
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

Learn more about defining variables.

easyC Pro ♦ Function Blocks ♦ Print to Screen

Use the Print to Screen block to display a message in the terminal window as the program is running.

Each Print to Screen function block will print out information in the terminal window on a single line (default) or multiple functions on the same line if Newline Character box is unchecked. The information can be a simple message (i.e. "Start program") or it can be a message followed by the value of a variable. Print to Screen is only designed to output a single variable. Users desiring a more robust output should consider writing a printf function in a User Code block.



- Use **Directives** to produce a custom output. The directive field is fully editable.

  Note
  This is an option for advanced users with C-programming experience. The default %d will work for most defined variables except long.

  Field Width

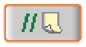
  Placing a number between the percent sign and the letter characters (called field width) will specify a minimum number of digits to be displayed, and will right justify (rather than left justify) the printed number. If the value printed is shorter than the field width, empty spaces will be inserted before the number. If the number is longer than the field width, the entire number will still be printed. Field width is generally used to create orderly columns, or to ensure a string has completely cleared when printing a shorter number to the same location.

  See an example

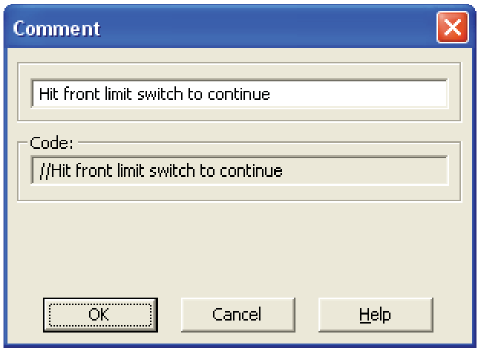
  See a chart of Print to screen options

- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

easyC Pro ♦ Function Blocks ♦ Comment



Use the Comment block to write notes in the program. Comments are ignored by the compiler. Use comments to make the program easier to understand.



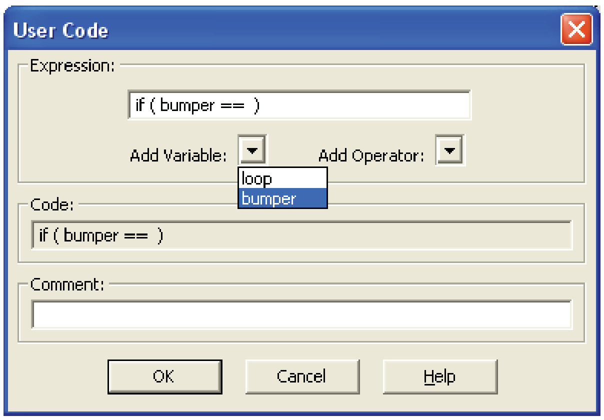
- In the Code area, view a preview of the C code that will be generated with the defined properties.

easyC Pro ♦ Function Blocks ♦ User Code

Use the User Code to add your own custom blocks of code to the program.

Note:
This feature is for advanced users familiar with C programming. You must use the correct syntax for the compiler to understand the code. Refer to a C-programming reference guide for information on C-programming.

The user code block is intended to be used for brief snippets of user code, such as function calls and print statements. Longer work should be undertaken from the project tab using source files.



- Use the drop down lists to select available variables and commonly used operators.

  View the Operator definition table.

- In the Code area, view a preview of the C code as its generated.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

Learn More:

Source Files

Function Blocks: RC Features

easyC Pro ♦ Function Blocks ♦ Arcade - 2 Motors

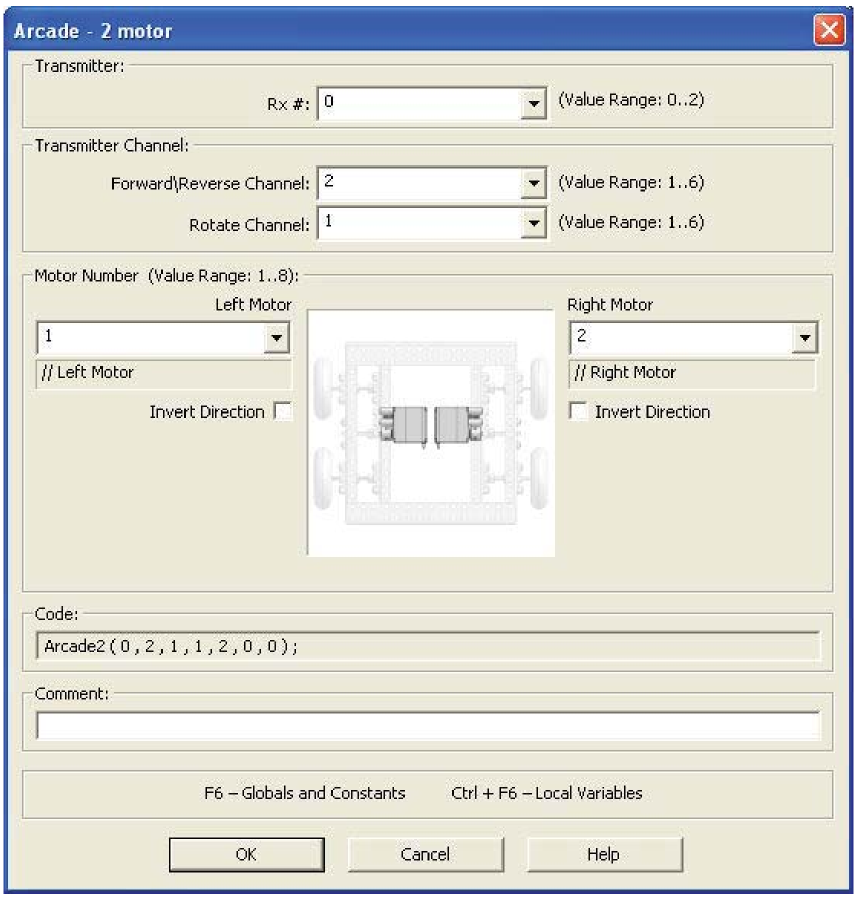
Use the Arcade – 2 motor function block to control a robot with 2 drive motors using an arcade style.

View an example of a program using RC control.

When using the RC function blocks, you must always place them in a loop (WHILE or FOR) to keep updating the controller with changes from the transmitter (Rx) as you control the robot.

The expressions "While (1)" and "While (1==1)" are called infinite loops or dead loops. These expressions are always true, so the loops will repeat forever. This is useful if you are doing a single sequence of events, like controlling the robot remotely.

Click to view a dead loop example



- Define the Transmitter Rx:

  - Select Rx# 1 to use the transmitter/receiver connected to Rx1 on the controller.
  - Select Rx# 2 to use the transmitter/receiver connected to Rx2 on the controller.
  - If you have only one transmitter and receiver, you can select Rx0, and the code will automatically select the connected receiver.

- Specify which channels on Rx 1 or Rx 2 that will be used as inputs to control the forward/reverse motion and rotation of your robot.
- Define the two motor outputs on the Vex controller that match your design.
  If necessary for your design, you can invert the direction of the motor by clicking the check box.

Note:
To use two transmitters to control your robot, you must use two different RF frequency crystals. The starter kit comes with RF frequency #61. More frequencies are available in the crystal upgrade kit.

- A description field lies below each Motor # combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

Caution
The transmitter must be set to 2-3 mode in order for this function to work properly. To reset your transmitter to its default 2-3 setting, hold down both the Mode and Select buttons on your transmitter until you hear a beep. Then using only the Mode button, cycle through the menus until you reach the 'Drive' menu, the last on the list. Use the Data Input button to switch modes if necessary.

View notes on Autonomous and competition mode

easyC Pro ♦ Function Blocks ♦ Arcade - 4 Motors

4

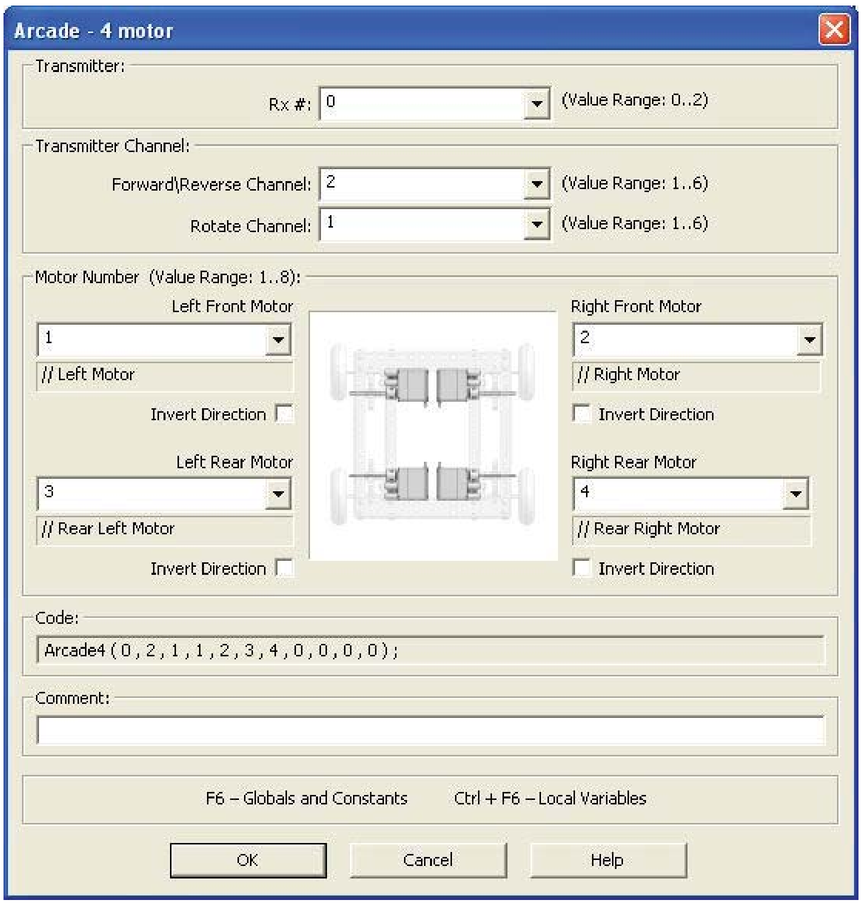
Use the Arcade – 4 motor function block to control a robot with 4 drive motors using an arcade style.

View an example of a generic program using RC control

When using the RC function blocks, you must always place them in a loop (WHILE or FOR) to keep updating the controller with changes from the transmitter (Rx) as you control the robot.

The expressions "While (1)" and "While (1==1)" are called infinite loops or dead loops. These expressions are always true, so the loops will repeat forever. This is useful if you are doing a single sequence of events, like controlling the robot remotely.

Click to view a dead loop example



- Define the Transmitter Rx:

- Select Rx# 1 to use the transmitter/receiver connected to Rx1 on the controller.
- Select Rx# 2 to use the transmitter/receiver connected to Rx2 on the controller.
- If you have only one transmitter and receiver, you can select Rx0, and the code will automatically select the connected receiver.

- Specify which channels on Rx 1 or Rx 2 that will be used as inputs to control the forward/reverse motion and rotation of your robot.
- Define the two motor outputs on the Vex controller that match your design.
  If necessary for your design, you can invert the direction of the motor by clicking the check box.

Note:
To use two transmitters to control your robot, you must use two different RF frequency crystals. The starter kit comes with RF frequency #61. More frequencies are available in the crystal upgrade kit.

- A description field lies below each Motor # combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

Caution
The transmitter must be set to 2-3 mode in order for this function to work properly. To reset your transmitter to its default 2-3 setting, hold down both the Mode and Select buttons on your transmitter until you hear a beep. Then using only the Mode button, cycle through the menus until you reach the 'Drive' menu, the last on the list. Use the Data Input button to switch modes if necessary.

View notes on Autonomous and competition mode

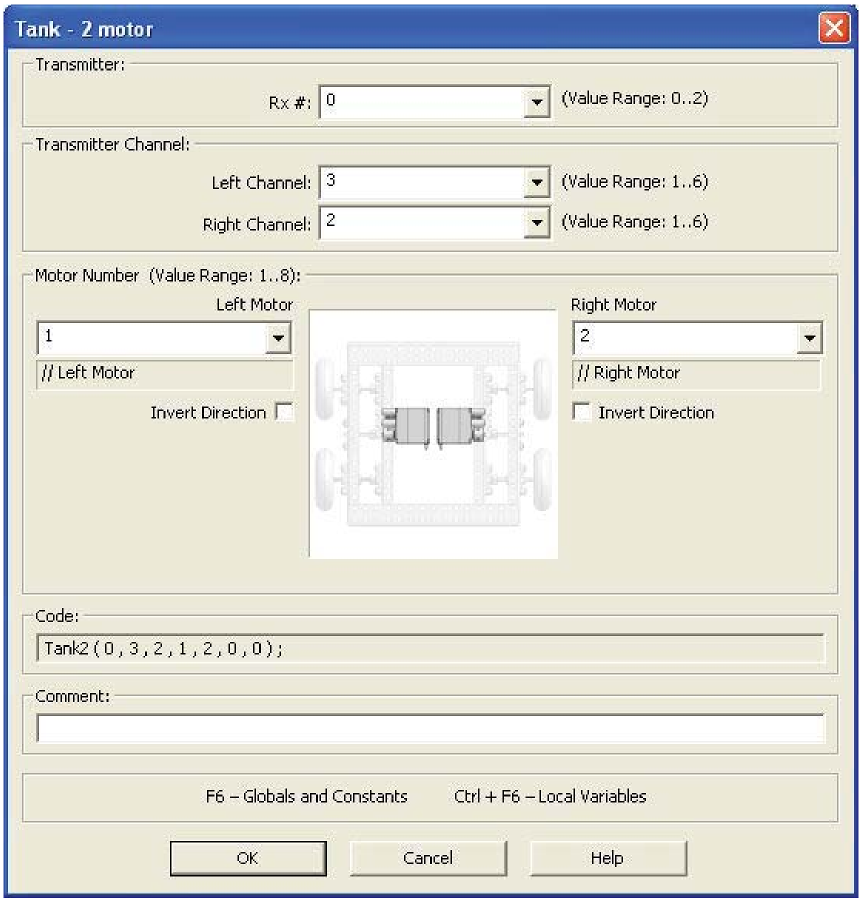
easyC Pro ♦ Function Blocks ♦ Tank - 2 Motor

Use the Tank - 2 motor function block to control a robot with 2 drive motors using a tank style.

When using the RC function blocks, you must always place them in a loop (WHILE or FOR) to keep updating the controller with changes from the transmitter (Rx) as you control the robot.

The expressions "While (1)" and "While (1==1)" are called infinite loops or dead loops. These expressions are always true, so the loops will repeat forever. This is useful if you are doing a single sequence of events, like controlling the robot remotely.

Click to view a dead loop example



- Define the Transmitter Rx:
  - Select Rx# 1 to use the transmitter/receiver connected to Rx1 on the controller.
  - Select Rx# 2 to use the transmitter/receiver connected to Rx2 on the controller.

- If you have only one transmitter and receiver, you can select Rx0, and the code will automatically select the connected receiver.

- Specify which channels on Rx 1 or Rx 2 that will be used as inputs to control the forward/reverse motion and rotation of your robot.
- Define the two motor outputs on the Vex controller that match your design.
  If necessary for your design, you can invert the direction of the motor by clicking the check box.

Note:
To use two transmitters to control your robot, you must use two different RF frequency crystals. The starter kit comes with RF frequency #61. More frequencies are available in the crystal upgrade kit.

- A description field lies below each Motor # combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

Caution
The transmitter must be set to 2-3 mode in order for this function to work properly. To reset your transmitter to its default 2-3 setting, hold down both the Mode and Select buttons on your transmitter until you hear a beep. Then using only the Mode button, cycle through the menus until you reach the 'Drive' menu, the last on the list. Use the Data Input button to switch modes if necessary.

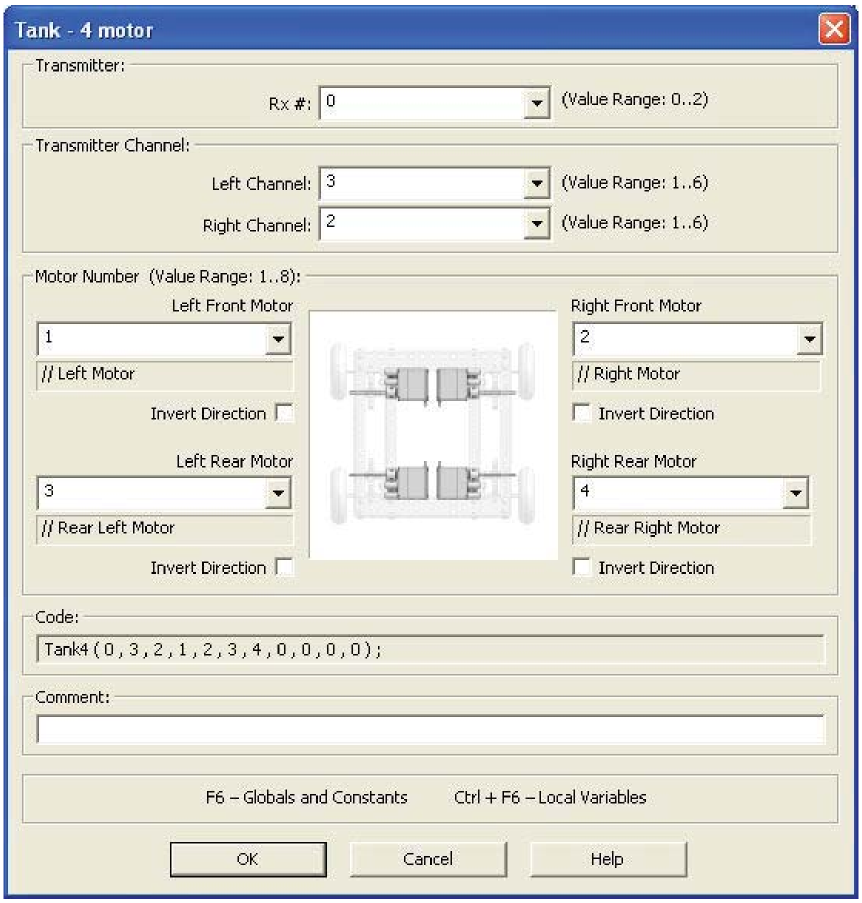
View notes on Autonomous and competition mode

easyC Pro ♦ Function Blocks ♦ Tank - 4 Motor

Use the Tank - 4 motor function block to control a robot with 4 drive motors using a tank style.

When using the RC function blocks, you must always place them in a loop (WHILE or FOR) to keep updating the controller with changes from the transmitter (Rx) as you control the robot.
The expressions "While (1)" and "While (1==1)" are called infinite loops or dead loops. These expressions are always true, so the loops will repeat forever. This is useful if you are doing a single sequence of events, like controlling the robot remotely.

Click to view a dead loop example



- Define the Transmitter Rx:
  - Select Rx# 1 to use the transmitter/receiver connected to Rx1 on the controller.
  - Select Rx# 2 to use the transmitter/receiver connected to Rx2 on the controller.

- If you have only one transmitter and receiver, you can select Rx0, and the code will automatically select the connected receiver.

- Specify which channels on Rx 1 or Rx 2 that will be used as inputs to control the forward/reverse motion and rotation of your robot.
- Define the four motor outputs on the Vex controller that match your design.
  If necessary for your design, you can invert the direction of the motor by clicking the check box.

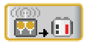
Note:
To use two transmitters to control your robot, you must use two different RF frequency crystals. The starter kit comes with RF frequency #61. More frequencies are available in the crystal upgrade kit.

- A description field lies below each Motor # combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

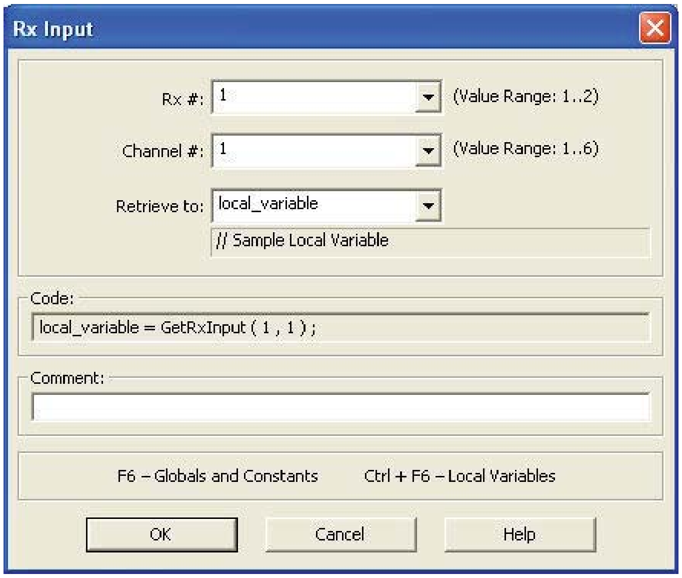
Caution
The transmitter must be set to 2-3 mode in order for this function to work properly. To reset your transmitter to its default 2-3 setting, hold down both the Mode and Select buttons on your transmitter until you hear a beep. Then using only the Mode button, cycle through the menus until you reach the 'Drive' menu, the last on the list. Use the Data Input button to switch modes if necessary.

View notes on Autonomous and competition mode

easyC Pro ♦ Function Blocks ♦ Rx Input



Use the Rx input function block to get feedback from one of the channels on Rx1 or Rx2 and store it in a variable.



- Define the Transmitter Rx:
  - Select Rx# 1 to use the transmitter/receiver connected to Rx1 on the controller.
  - Select Rx# 2 to use the transmitter/receiver connected to Rx2 on the controller.
  - If you have only one transmitter and receiver, you can select Rx0, and the code will automatically select the connected receiver.

Note:
To use two transmitters to control your robot, you must use two different RF frequency crystals. The starter kit comes with RF frequency #61. More frequencies are available in the crystal upgrade kit.

- Select a predefined variable in the "Retrieve to" field. Alternatively, you can enter your own variable name now, in which case you must still define the variable as either local or global prior to compiling. You may hit either F6, or Ctrl + F6 at any time to open the variable definition windows.
- A description field lies below the 'Retrieve to' combo box. The description field will display comments and descriptions from the selection above, should any exist.
- In the Code area, view a preview of the C code that will be generated with the defined properties.
- In the Comment field, enter comments that will help you read the program and understand the function of the block without knowing all the properties defined by the block.

- Open RXINPUT.ecp found in the Test Code folder to see a sample program which manipulates a value taken directly from the transmitter before sending a command to the motors.

Caution

The transmitter must be set to 2-3 mode in order for this function to work properly. To reset your transmitter to its default 2-3 setting, hold down both the Mode and Select buttons on your transmitter until you hear a beep. Then using only the Mode button, cycle through the menus until you reach the 'Drive' menu, the last on the list. Use the Data Input button to switch modes if necessary.