

## **TEACHER'S GUIDE**

**ROBOTICS**  
Building System



# TEACHER'S GUIDE

## ROBOTICS Building System

### TABLE OF CONTENTS:

#### ALIGNMENTS

COMPUTING .....	3
DESIGN AND TECHNOLOGY .....	4
ENGLISH.....	5
MATHEMATICS .....	7
SCIENCE .....	11

INTRODUCTION .....	14
--------------------	----

#### LESSON 1

Motor Control – Line Tracker.....	17
-----------------------------------	----

#### LESSON 2

Line Tracker .....	28
--------------------	----

#### LESSON 3

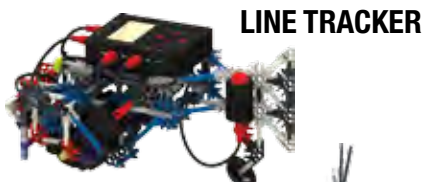
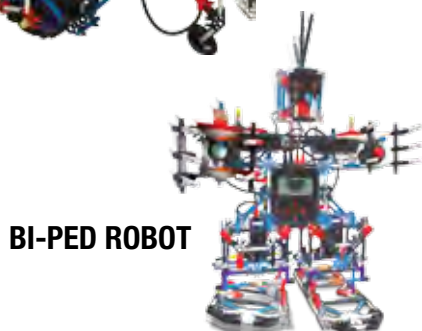
Bi-Ped Robot.....	37
-------------------	----

#### LESSON 4

Search & Rescue Robot.....	51
----------------------------	----

#### LESSON 5

Ball Factory .....	59
--------------------	----

**LINE TRACKER****BI-PED ROBOT****SEARCH &  
RESCUE ROBOT****BALL FACTORY**

**WARNING:** CHOKING HAZARD – Small parts.

Not for children under 3 years.

### A NOTE ABOUT SAFETY:

Safety is of primary concern in science and technology classrooms. It is recommended that you develop a set of rules that governs the safe, proper use of K'NEX in your classroom. Caution students to keep hands, face, hair and clothing away from all moving parts.

# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to The National Curriculum in England – Computing

*Computing Overview: K'NEX Robotics covers all the requirements of the Computing National Curriculum and the three areas of Computer Science, Information Technology and Digital Literacy.*

## **At KEY STAGE 2 pupils should be taught to:**

- Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts.
- Use sequence, selection, and repetition in programs; work with variables and various forms of input and output.
- Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs.
- Understand computer networks, including the internet; how they can provide multiple services, such as the World Wide Web, and the opportunities they offer for communication and collaboration.
- Use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content.
- Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information.
- Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact.

## **At KEY STAGE 3 pupils should be taught to:**

- Design, use and evaluate computational abstractions that model the state and behaviour of real-world problems and physical systems.
- Understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem.
- Use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures [for example, lists, tables or

arrays]; design and develop modular programs that use procedures or functions.

- Understand simple Boolean logic [for example, AND, OR and NOT] and some of its uses in circuits and programming; understand how numbers can be represented in binary, and be able to carry out simple operations on binary numbers [for example, binary addition, and conversion between binary and decimal].
- Understand the hardware and software components that make up computer systems, and how they communicate with one another and with other systems.
- Understand how instructions are stored and executed within a computer system; understand how data of various types (including text, sounds and pictures) can be represented and manipulated digitally, in the form of binary digits.
- Undertake creative projects that involve selecting, using, and combining multiple applications, preferably across a range of devices, to achieve challenging goals, including collecting and analysing data and meeting the needs of known users.
- Create, reuse, revise and repurpose digital artefacts for a given audience, with attention to trustworthiness, design and usability.
- Understand a range of ways to use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy; recognise inappropriate content, contact and conduct, and know how to report concerns.

## **At KEY STAGE 4 all pupils should be taught to:**

- Develop their capability, creativity and knowledge in computer science, digital media and information technology.
- Develop and apply their analytic, problem-solving, design, and computational thinking skills.
- Understand how changes in technology affect safety, including new ways to protect their online privacy and identity, and how to report a range of concerns.



# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to the National Curriculum in England – Design and Technology

## KEY STAGE 2

*When designing and making, pupils should be taught to:*

### Design

- Use research and develop design criteria to inform the design of innovative, functional, appealing products that are fit for purpose, aimed at particular individuals or groups.
- Generate, develop, model and communicate their ideas through discussion, annotated sketches, cross-sectional and exploded diagrams, prototypes, pattern pieces and computer-aided design.

### Make

- Select from and use a wider range of tools and equipment to perform practical tasks [for example, cutting, shaping, joining and finishing], accurately.
- Select from and use a wider range of materials and components, including construction materials, textiles and ingredients, according to their functional properties and aesthetic qualities.

### Evaluate

- Investigate and analyse a range of existing products.
- Evaluate their ideas and products against their own design criteria and consider the views of others to improve their work.
- Understand how key events and individuals in design and technology have helped shape the world.

### Technical Knowledge

- Apply their understanding of how to strengthen, stiffen and reinforce more complex structures.
- Understand and use mechanical systems in their products [for example, gears, pulleys, cams, levers and linkages].
- Understand and use electrical systems in their products [for example, series circuits incorporating switches, bulbs, buzzers and motors].
- Apply their understanding of computing to program, monitor and control their products.

## KEY STAGE 3

*When designing and making, pupils should be taught to:*

### Design

- Use research and exploration, such as the study of different cultures, to identify and understand user needs.

- Identify and solve their own design problems and understand how to reformulate problems given to them.
- Develop specifications to inform the design of innovative, functional, appealing products that respond to needs in a variety of situations.
- Use a variety of approaches [for example, biomimicry and user-centred design] to generate creative ideas and avoid stereotypical responses.
- Develop and communicate design ideas using annotated sketches, detailed plans, 3-D and mathematical modelling, oral and digital presentations.

### Make

- Select from and use specialist tools, techniques, processes, equipment and machinery precisely, including computer-aided manufacture.
- Select from and use a wider, more complex range of materials, components and ingredients, taking into account their properties.

### Evaluate

- Analyse the work of past and present professionals and others to develop and broaden their understanding.
- Investigate new and emerging technologies.
- Test, evaluate and refine their ideas and products against a specification, taking into account the views of intended users and other interested groups.
- Understand developments in design and technology, its impact on individuals, society and the environment, and the responsibilities of designers, engineers and technologists.

### Technical Knowledge

- Understand and use the properties of materials and the performance of structural elements to achieve functioning solutions.
- Understand how more advanced mechanical systems used in their products enable changes in movement and force.
- Understand how more advanced electrical and electronic systems can be powered and used in their products [for example, circuits with heat, light, sound and movement as inputs and outputs].
- Apply computing and use electronics to embed intelligence in products that respond to inputs [for example, sensors] and control outputs [for example, actuators] using programmable components [for example, microcontrollers].

# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide – English

## KS2 Literacy

### Spoken language (Y1-6)

- Ask relevant questions to extend their understanding and knowledge.
- Articulate and justify answers, arguments and opinions.
- Give well-structured descriptions, explanations and narratives for different purposes, including for expressing feelings.
- Maintain attention and participate actively in collaborative conversations, staying on topic and initiating and responding to comments.
- Use spoken language to develop understanding through speculating, hypothesising, imagining and exploring ideas.
- Participate in discussions, presentations, performances, role play, improvisations and debates.
- Consider and evaluate different viewpoints, attending to and building on the contributions of others.

### KS2 Reading

*Understand what they read by:*

- Asking questions to improve their understanding (Y5-6).
- Provide reasoned justifications for their views (Y5-6).
- Retrieve, record and present information from non-fiction (Y5-6).
- Provide reasoned justifications for their views (Y5-6).

### KS2 Writing

*Writing – transcription*

- Use dictionaries to check the spelling and meaning of words.

*Writing – composition*

Plan their writing by:

- Identifying the audience for and purpose of the writing, selecting the appropriate form and using other similar writing as models for their own (Y5-6).
- Noting and developing initial ideas, drawing on reading and research where necessary (Y5-6).

## KS3 Literacy

### Reading

*Understand increasingly challenging texts through:*

- Learning new vocabulary, relating it explicitly to known vocabulary and understanding it with the help of context and dictionaries.
- Knowing the purpose, audience for and context of

the writing and drawing on this knowledge to support comprehension.

- Checking their understanding to make sure that what they have read makes sense.

### Writing

*Write accurately, fluently, effectively and at length for pleasure and information through:*

- Writing for a wide range of purposes and audiences, including:
- Notes and polished scripts for talks and presentations.
- Summarising and organising material, and supporting ideas and arguments with any necessary factual detail.
- Applying their growing knowledge of vocabulary, grammar and text structure to their writing and selecting the appropriate form.
- Plan, draft, edit and proof-read through:
- Considering how their writing reflects the audiences and purposes for which it was intended.
- Amending the vocabulary, grammar and structure of their writing to improve its coherence and overall effectiveness.
- Paying attention to accurate grammar, punctuation and spelling; applying the spelling patterns and rules set out in English Appendix 1 to the key stage 1 and 2 programmes of study for English.

### Spoken English

*Speak confidently and effectively, including through:*

- Using Standard English confidently in a range of formal and informal contexts, including classroom discussion.
- Giving short speeches and presentations, expressing their own ideas and keeping to the point.
- Participating in formal debates and structured discussions, summarising and/or building on what has been said.
- Improvising, rehearsing and performing play scripts and poetry in order to generate language and discuss language use and meaning, using role, intonation.
- Tone, volume, mood, silence, stillness and action to add impact.
- Use discussion in order to learn; they should be able to elaborate and explain clearly their understanding and ideas.
- Are competent in the arts of speaking and listening, making formal presentations, demonstrating to others and participating in debate.

# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide – English

## KS4 Literacy

### Reading

*Understand and critically evaluate texts through:*

- Distinguishing between statements that are supported by evidence and those that are not, and identifying bias and misuse of evidence.

### Writing

*Write accurately, fluently, effectively and at length for pleasure and information through:*

- Adapting their writing for a wide range of purposes and audiences: to describe.
- Narrate, explain, instruct, give and respond to information, and argue.
- Make notes, draft and write, including using information provided by others [e.g. writing a letter from key points provided; drawing on and using information from a presentation].

*Revise, edit and proof-read through:*

- Reflecting on whether their draft achieves the intended impact.
- Restructuring their writing, and amending its grammar and vocabulary to improve coherence, consistency, clarity and overall effectiveness.

## Spoken English

*Speak confidently, audibly and effectively, including through:*

- Working effectively in groups of different sizes and taking on required roles, including leading and managing discussions, involving others productively, reviewing and summarising, and contributing to meeting goals/deadlines.
- Listening to and building on the contributions of others, asking questions to clarify and inform, and challenging courteously when necessary.
- Planning for different purposes and audiences, including selecting and organising information and ideas effectively and persuasively for formal spoken presentations and debates.
- Listening and responding in a variety of different contexts, both formal and informal, and evaluating content, viewpoints, evidence and aspects of presentation.
- Improvising, rehearsing and performing play scripts and poetry in order to generate language and discuss language use and meaning, using role, intonation, tone, volume, mood, silence, stillness and action to add impact.

# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to the UK National Curriculum – Mathematics

## Mathematical aims for all pupils in KS2

- Become fluent in the fundamentals of mathematics, including through varied and frequent practice with increasingly complex problems over time, so that pupils develop conceptual understanding and the ability to recall and apply knowledge rapidly and accurately.
- Reason mathematically by following a line of enquiry, conjecturing relationships and generalisations, and developing an argument, justification or proof using mathematical language.
- Can solve problems by applying their mathematics to a variety of routine and non-routine problems with increasing sophistication, including breaking down problems into a series of simpler steps and persevering in seeking solutions.

## YEAR 5

### Number – Multiplication and Division

- Identify multiples and factors, including finding all factor pairs of a number, and common factors of two numbers.

### Number – Fractions (including decimals and percentages)

- Recognise and use thousandths and relate them to tenths, hundredths and decimal equivalents.
- Round decimals with two decimal places to the nearest whole number and to one decimal place.
- Read, write, order and compare numbers with up to three decimal places.
- Solve problems involving number up to three decimal places.
- Recognise the per cent symbol (%) and understand that per cent relates to 'number of parts per hundred', and write percentages as a fraction with denominator 100, and as a decimal.

### Measurement

- Convert between different units of metric measure (for example, kilometre and metre; centimetre and metre; centimetre and millimetre; gram and kilogram; litre and millilitre).
- Use all four operations to solve problems involving measure [for example, length, mass, volume, money] using decimal notation, including scaling.

### Geometry – Properties of Shapes

- Know angles are measured in degrees: estimate and compare acute, obtuse and reflex angles.

- Draw given angles, and measure them in degrees.
- Identify angles at a point and one whole turn, angles at a point on a straight line and  $\frac{1}{2}$  a turn and other multiples of 90 degrees.

### Geometry – Position and Direction

- Identify, describe and represent the position of a shape following a reflection or translation, using the appropriate language, and know that the shape has not changed.

### Statistics

- Complete, read and interpret information in tables, including timetables.

## YEAR 6

### Number – Addition, Subtraction, Multiplication and Division

- Perform mental calculations, including with mixed operations and large numbers.
- Identify common factors, common multiples and prime numbers.
- Use their knowledge of the order of operations to carry out calculations involving the four operations.
- Solve problems involving addition, subtraction, multiplication and division.
- Use estimation to check answers to calculations and determine, in the context of a problem, an appropriate degree of accuracy.

### Number – Fractions (including decimals and percentages)

- Identify the value of each digit in numbers given to three decimal places and multiply and divide numbers by 10, 100 and 1000 giving answers up to three decimal places.
- Solve problems which require answers to be rounded to specified degrees of accuracy.
- Recall and use equivalences between simple fractions, decimals and percentages, including in different contexts.

### Ratio and Proportion

- Solve problems involving the calculation of percentages [for example, of measures, and such as 15% of 360] and the use of percentages for comparison.

### Algebra

- Use simple formulae.
- Enumerate possibilities of combinations of two variables.

# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to the UK National Curriculum – Mathematics

## Measurement

- Solve problems involving the calculation and conversion of units of measure, using decimal notation up to three decimal places where appropriate.
- Use, read, write and convert between standard units, converting measurements of length, mass, volume and time from a smaller unit of measure to a larger unit, and vice versa, using decimal notation to up to three decimal places.

## Statistics

- Interpret and construct pie charts and line graphs and use these to solve problems.

## At KEY STAGE 3 pupils are encouraged to work mathematically and should be taught to:

### Develop fluency

- Consolidate their numerical and mathematical capability from key stage 2 and extend their understanding of the number system and place value to include decimals, fractions, powers and roots.
- Select and use appropriate calculation strategies to solve increasingly complex problems.
- Use algebra to generalise the structure of arithmetic, including to formulate mathematical relationships.
- Substitute values in expressions, rearrange and simplify expressions, and solve equations.
- Move freely between different numerical, algebraic, graphical and diagrammatic representations [for example, equivalent fractions, fractions and decimals, and equations and graphs].
- Develop algebraic and graphical fluency, including understanding linear and simple quadratic functions.
- Use language and properties precisely to analyse numbers, algebraic expressions, 2-D and 3-D shapes, probability and statistics.

### Reason Mathematically

- Extend their understanding of the number system; make connections between number relationships, and their algebraic and graphical representations
- Extend and formalise their knowledge of ratio and proportion in working with measures and geometry, and in formulating proportional relations algebraically.
- Identify variables and express relations between variables algebraically and graphically.

- Make and test conjectures about patterns and relationships; look for proofs or counter-examples.
- Begin to reason deductively in geometry, number and algebra, including using geometrical constructions.
- Interpret when the structure of a numerical problem requires additive, multiplicative or proportional reasoning.
- Explore what can and cannot be inferred in statistical and probabilistic settings, and begin to express their arguments formally.

### Solve Problems

- Develop their mathematical knowledge, in part through solving problems and evaluating the outcomes, including multi-step problems.
- Develop their use of formal mathematical knowledge to interpret and solve problems, including in financial mathematics.
- Begin to model situations mathematically and express the results using a range of formal mathematical representations.
- Select appropriate concepts, methods and techniques to apply to unfamiliar and non-routine problems.

### Number

- Order positive and negative integers, decimals and fractions; use the number line as a model for ordering of the real numbers; use the symbols  $=$ ,  $\neq$ ,  $<$ ,  $\leq$ ,  $>$ ,  $\geq$ .
- Understand and use place value for decimals, measures and integers of any size.
- Use the concepts and vocabulary of prime numbers, factors (or divisors), multiples, common factors, common multiples, highest common factor, lowest common multiple, prime factorisation, including using product notation and the unique factorisation property.
- Use the four operations, including formal written methods, applied to integers, decimals, proper and improper fractions, and mixed numbers, all both positive and negative.
- Define percentage as 'number of parts per hundred', interpret percentages and percentage changes as a fraction or a decimal, interpret these multiplicatively, express one quantity as a percentage of another, compare two quantities using percentages, and work with percentages greater than 100%.
- Use standard units of mass, length, time, money and other measures, including with decimal quantities.



# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to the UK National Curriculum – Mathematics

- Round numbers and measures to an appropriate degree of accuracy [for example, to a number of decimal places or significant figures]

## Algebra

- Substitute numerical values into formulae and expressions, including scientific formulae.
- Understand and use the concepts and vocabulary of expressions, equations, inequalities, terms and factors.
- Understand and use standard mathematical formulae; rearrange formulae to change the subject.
- Model situations or procedures by translating them into algebraic expressions or formulae and by using graphs.
- Use algebraic methods to solve linear equations in one variable (including all forms that require rearrangement).
- Recognise, sketch and produce graphs of linear and quadratic functions of one variable with appropriate scaling, using equations in  $x$  and  $y$  and the Cartesian plane.
- Interpret mathematical relationships both algebraically and graphically.

## Ratio, Proportion and Rates of Change

- Change freely between related standard units [for example time, length, area, volume/capacity, mass].
- Solve problems involving percentage change, including: percentage increase, decrease and original value problems and simple interest in financial mathematics.

## Geometry and Measures

- Identify properties of, and describe the results of, translations, rotations and reflections applied to given figures.
- Interpret mathematical relationships both algebraically and geometrically.

## Statistics

- Describe, interpret and compare observed distributions of a single variable through: appropriate graphical representation involving discrete, continuous and grouped data; and appropriate measures of central tendency (mean, mode, median) and spread (range, consideration of outliers).
- Construct and interpret appropriate tables, charts, and diagrams, including frequency tables, bar charts, pie charts, and pictograms for categorical data, and vertical line (or bar) charts for ungrouped and grouped numerical data.

**At KEY STAGE 4 pupils are encouraged to work mathematically and should be taught to:**

### Develop Fluency

- Consolidate their numerical and mathematical capability from key stage 3 and extend their understanding of the number system to include powers, roots {and fractional indices}
- Select and use appropriate calculation strategies to solve increasingly complex problems, including exact calculations involving multiples of  $\pi$  {and surds}, use of standard form and application and interpretation of limits of accuracy.
- Consolidate their algebraic capability from key stage 3 and extend their understanding of algebraic simplification and manipulation to include quadratic expressions, {and expressions involving surds and algebraic fractions}.
- Move freely between different numerical, algebraic, graphical and diagrammatic representations, including of linear, quadratic, reciprocal, {exponential and trigonometric} functions.
- Use mathematical language and properties precisely.

### Reason Mathematically

- Extend and formalise their knowledge of ratio and proportion, including trigonometric ratios, in working with measures and geometry, and in working with proportional relations algebraically and graphically.
- Extend their ability to identify variables and express relations between variables algebraically and graphically.
- Make and test conjectures about the generalisations that underlie patterns and relationships; look for proofs or counter-examples; begin to use algebra to support and construct arguments {and proofs}.
- Reason deductively in geometry, number and algebra, including using geometrical constructions.
- Interpret when the structure of a numerical problem requires additive, multiplicative or proportional reasoning.
- Assess the validity of an argument and the accuracy of a given way of presenting information.

### Solve Problems

- Develop their mathematical knowledge, in part through solving problems and evaluating the outcomes, including multi-step problems.

# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to the UK National Curriculum – Mathematics

- Develop their use of formal mathematical knowledge to interpret and solve problems, including in financial contexts.
- Make and use connections between different parts of mathematics to solve problems.
- Model situations mathematically and express the results using a range of formal mathematical representations, reflecting on how their solutions may have been affected by any modelling assumptions.
- Select appropriate concepts, methods and techniques to apply to unfamiliar and non-routine problems; interpret their solution in the context of the given problem.

## Algebra

- Where appropriate, interpret simple expressions as functions with inputs and outputs.
- Plot and interpret graphs (including reciprocal graphs {and exponential graphs}) and graphs of non-standard functions in real contexts, to find approximate solutions to problems such as simple kinematic problems involving distance, speed and acceleration.

# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to The National Curriculum in England – Science

## KEY STAGE 2 – Working Scientifically

During years 5 and 6, pupils should be taught to use the following practical scientific methods, processes and skills through the teaching of the programme of study content:

- Planning different types of scientific enquiries to answer questions, including recognising and controlling variables where necessary.
- Taking measurements, using a range of scientific equipment, with increasing accuracy and precision, taking repeat readings when appropriate.
- Recording data and results of increasing complexity using scientific diagrams and labels, classification keys, tables, scatter graphs, bar and line graphs.
- Using test results to make predictions to set up further comparative and fair tests.
- Reporting and presenting findings from enquiries, including conclusions, causal relationships and explanations of and a degree of trust in results, in oral and written forms such as displays and other presentations.
- Identifying scientific evidence that has been used to support or refute ideas or arguments.

## YEAR 6

### Electricity

*Pupils should be taught to:*

- Associate the brightness of a lamp or the volume of a buzzer with the number and voltage of cells used in the circuit.
- Compare and give reasons for variations in how components function, including the brightness of bulbs, the loudness of buzzers and the on/off position of switches.
- Use recognised symbols when representing a simple circuit in a diagram.

### Notes and guidance (non-statutory)

Building on their work in year 4, pupils should construct simple series circuits, to help them to answer questions about what happens when they try different components, for example, switches, bulbs, buzzers and motors. They should learn how to represent a simple circuit in a diagram using recognised symbols.

## KEY STAGE 3 – Working Scientifically

*Through the content across all three disciplines, pupils should be taught to:*

### Scientific attitudes

- Pay attention to objectivity and concern for accuracy, precision, repeatability and reproducibility.

- Understand that scientific methods and theories develop as earlier explanations are modified to take account of new evidence and ideas, together with the importance of publishing results and peer review.
- Evaluate risks.

### Experimental skills and Investigations

- Ask questions and develop a line of enquiry based on observations of the real world, alongside prior knowledge and experience.
- Make predictions using scientific knowledge and understanding.
- Select, plan and carry out the most appropriate types of scientific enquiries to test predictions, including identifying independent, dependent and control variables.
- Use appropriate techniques, apparatus, and materials during fieldwork and laboratory work, paying attention to health and safety.
- Make and record observations and measurements using a range of methods for different investigations; and evaluate the reliability of methods and suggest possible improvements.
- Apply sampling techniques.

### Analysis and Evaluation

- Apply mathematical concepts and calculate results.
- Present observations and data using appropriate methods, including tables and graphs.
- Interpret observations and data, including identifying patterns and using observations, measurements and data to draw conclusions.
- Present reasoned explanations, including explaining data in relation to predictions and hypotheses.
- Evaluate data, showing awareness of potential sources of random and systematic error.
- Identify further questions arising from their results.

### Measurement

- Use and derive simple equations and carry out appropriate calculations
- Undertake basic data analysis including simple statistical techniques

## Physics

### Motion and Forces

#### Describing Motion

- Speed and the quantitative relationship between average speed, distance and time (speed = distance ÷ time)

# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to The National Curriculum in England – Science

- The representation of a journey on a distance-time graph.
- Relative motion: trains and cars passing one another.

## Forces

- Forces as pushes or pulls, arising from the interaction between 2 objects.
- Using force arrows in diagrams, adding forces in 1 dimension, balanced and unbalanced forces.
- Moment as the turning effect of a force.
- Forces: associated with deforming objects; stretching and squashing – springs; with rubbing and friction between surfaces, with pushing things out of the way; resistance to motion of air and water.
- Non-contact forces: gravity forces acting at a distance on Earth and in space, forces between magnets, and forces due to static electricity.

## Forces and Motion

- Forces being needed to cause objects to stop or start moving, or to change their speed or direction of motion (qualitative only).
- Change depending on direction of force and its size.

## KEY STAGE 4 – Working Scientifically

### 1. The development of scientific thinking

- The ways in which scientific methods and theories develop over time.
- Using a variety of concepts and models to develop scientific explanations and understanding.
- Appreciating the power and limitations of science and considering ethical issues which may arise.
- Explaining everyday and technological applications of science; evaluating associated personal, social, economic and environmental implications; and making decisions based on the evaluation of evidence and arguments.
- Evaluating risks both in practical science and the wider societal context, including perception of risk.
- Recognising the importance of peer review of results and of communication of results to a range of audiences.

### 2. Experimental skills and strategies

- Using scientific theories and explanations to develop hypotheses.
- Planning experiments to make observations, test hypotheses or explore phenomena.
- Applying a knowledge of a range of techniques, apparatus, and materials to select those appropriate both for fieldwork and for experiments.

- Carrying out experiments appropriately, having due regard to the correct manipulation of apparatus, the accuracy of measurements and health and safety considerations.
- Recognising when to apply a knowledge of sampling techniques to ensure any samples collected are representative.
- Making and recording observations and measurements using a range of apparatus and methods.
- Evaluating methods and suggesting possible improvements and further investigations.

### 3. Analysis and evaluation

- Applying the cycle of collecting, presenting and analysing data, including:
- Presenting observations and other data using appropriate methods.
- Translating data from one form to another.
- Carrying out and representing mathematical and statistical analysis.
- Representing distributions of results and making estimations of uncertainty.
- Interpreting observations and other data, including identifying patterns and trends, making inferences and drawing conclusions.
- Presenting reasoned explanations, including relating data to hypotheses.
- Being objective, evaluating data in terms of accuracy, precision, repeatability and reproducibility and identifying potential sources of random and systematic error.
- Communicating the scientific rationale for investigations, including the methods used, the findings and reasoned conclusions, using paper-based and electronic reports and presentations.

### 4. Vocabulary, units, symbols and nomenclature

- Developing their use of scientific vocabulary and nomenclature.
- Recognising the importance of scientific quantities and understanding how they are determined.
- Using prefixes and powers of ten for orders of magnitude (e.g. tera, giga, mega, kilo, centi, milli, micro and nano).
- Interconverting units.
- Using an appropriate number of significant figures in calculations.



# Alignments of the K'NEX Education Robotics Building System Set and Teacher's Guide to The National Curriculum in England – Science

## Biology

### Coordination and Control

- Principles of nervous coordination and control in humans.

### Ecosystems

- Organisms are interdependent and are adapted to their environment.
- Positive and negative human interactions with ecosystems.

### Evolution, Inheritance and Variation

- The evidence for evolution.

## Physics

### Energy

- Energy changes in a system involving heating, doing work using forces, or doing work using an electric current: calculating the stored energies and energy changes involved.
- Power as the rate of transfer of energy.
- Conservation of energy in a closed system, dissipation.

- Calculating energy efficiency for any energy transfers.

- Renewable and non-renewable energy sources used on Earth, changes in how these are used.

### Forces

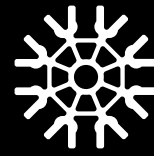
- Forces and fields: electrostatic, magnetic, gravity.
- Forces as vectors.
- Calculating work done as force x distance; elastic and inelastic stretching.
- Pressure in fluids acts in all directions: variation in Earth's atmosphere with height, with depth for liquids, up-thrust force (qualitative).

### Forces and Motion

- Speed of sound, estimating speeds and accelerations in everyday contexts.
- Interpreting quantitatively graphs of distance, time, and speed.
- Acceleration caused by forces; Newton's First Law.
- Weight and gravitational field strength.



# Introduction to K'NEX ROBOTICS



**W**elcome to the wonderful world of robotics with the K'NEX Robotics Set designed to meet the needs of students in a high interest STEAM program. Most will agree that robotics is the first thing that comes to mind when discussions turn to STEAM. This set will provide you or you and your students with exciting, large, fully functioning K'NEX models to build. In addition, the set includes easy to use software and custom designed electronics that will enable students to explore programming from the simple to the complex. As countries around the world seek to provide computer education and coding opportunities for all students, K'NEX has developed this comprehensive robotics set to help you and your students to meet this challenging goal.

A comprehensive Teacher's Guide, a Hardware User's Guide, and a Software User's Guide are provided to enhance your students' understanding of programming logic, structure and design. The software has been designed around a "Click and Drop" system using programmable output, process, and decision symbols. As the symbols are sequenced, they produce flowcharts to control any of the nine models in the K'NEX Robotics Set. Additionally, the flowcharts can be viewed in text form with a click of the mouse. The set includes: 4 motors, an LED, Distance Sensor, Colour/Light Sensor, Push Button Sensor, cables/wiring, 825 K'NEX pieces, Hardware User's Guide, Software User's Guide, and the K'NEX Robotics Control Box (*with battery power option*).

## **The Teacher's Guide:**

The Teacher's Guide includes five lessons which use four of the nine models that can be constructed from the set. The lessons progress from simple to complex programming as they introduce students to: the model's mechanical systems, motor operation, sensor operation and input, programming structure and design, and feedback systems. Initial lessons introduce basic programming strategies and include challenge activities to pique students' interest and desire to know and understand more! Later lessons enable students to explore the programming of machines with multiple motors, multiple sensor inputs, and output devices. These later lessons will encourage complex computational thinking and enable students to reach higher levels of programming excellence. Once students have completed the 5 lessons in the Teacher's Guide, they can move on to the other models in the K'NEX Robotics Set and let their imaginations guide their explorations as they expand their programming design skills in conjunction with the valuable information found in the K'NEX Hardware and Software User's Guides. Each lesson is aligned to the updated National curriculum in England.

Each lesson provides information for the teacher, instructional suggestions, objectives, materials list, and appropriate programming examples. Each lesson also provides Student Instruction Sheets and Student Response Sheets. The Student Instruction Sheets outline the lesson for the students so they can tackle the particular project or challenge on their own or with a minimum of supervision. The initial lessons will require more support for the students if they are new to programming. The models highlighted in the lessons are the Line Tracker (*used in two lessons*), Bi-Ped Robot, Search & Rescue Robot, and the Ball Factory. The Student Response Sheets provide charts and questions for the students to complete as they work through each lesson. These sheets enable students to keep a record of their work and they are excellent for the collection of assessment data.

## What Is Robotics?

A robot is essentially a mechanical device that has a number of input and output devices with a control system allowing intelligent decisions to be made depending on the feedback received. A robot is programmed to follow a set of instructions. It has a processing unit, sensors to perceive its environment, and motors and actuators to move its wheels or limbs. The robot may speak or generate other sounds, or it may flash with lights and colours in response to the environment according to the instructions given.

Programming is an abstract concept that robotics makes understandable and approachable. Everyone needs to learn how to program a computer and programming a robot is great fun that teaches students to adopt an engineering mind-set and a step-by-step way of making things work. Robots need the ability to follow programmed instructions and so robotics is a highly effective way of introducing programming to students. By building and controlling a physical robot students can see what works and what doesn't and they soon begin to appreciate the need for precise instructions: it is both challenging and creative. Robotics is a concrete and tangible way to build and strengthen cognitive development.

Robots are suited to all learners but they have been found to suit learners on the autism spectrum as they respond to the clear, calm and consistent interactions robots can provide.

A robotics lesson is a true inter-disciplinary experience because it provides students with an opportunity to learn about interaction, computer programming, physics, social and ethical dimensions, engineering, as well as persistence, curiosity, problem-solving and working in teams. Writing project journals, technical manuals, documentation, or basic instructions for classmates can all be incorporated into a robotics project and the aesthetics which go into the design and creation of robots allow students to experiment with artistic endeavours too. Not only do students learn to control a robot, they also get a more complete idea what is required for an organism to act in the real world. Robotics is a perfect way to knit computing into the design and technology process, merging both technical and creative skills.

## Why study Robotics?

Using the K'NEX Robotics set allows for the development of different skillsets and teachers have an opportunity to develop a wide range of skills among their students that will support them in their career beyond the classroom. For example, being able to code is going to be very important for a range of STEAM-based careers in the future, but it needs to be connected to something physical, such as programming a robot, to ground it in reality.

Robotics are cutting-edge and exciting and offer the perfect outlet for students because it:

- Offers a hands-on learning approach which students love.
- Incorporates multiple disciplines and opens career possibilities.
- Allows students to actively become involved in their own learning.
- Encourages active problem solving.
- Builds logical reasoning, analytical reasoning and critical thinking.
- Improves decision making.
- Demystifies programming.
- Engages students in computational thinking and higher order thinking.
- Promotes technological literacy.
- Students must demonstrate resource and time management.
- Encourages teamwork and problem solving.
- Can build engineering intuition.

Robotics strengthen and support students skills by developing their knowledge and understanding through the creation, design, assembly and operation of robots. Learning robotics makes students feel empowered, creative and confident.

K'NEX Robotics allows your students to be actively involved in technology. Why not create a robot lab where students can explore building and programming in more depth?

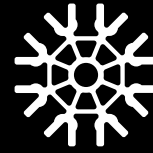
The teaching of robotics is a superb way to inject more fun into the curriculum, as well as providing a sound starting point for the future generation of designers, engineers and manufacturers. Robots provide a wholesome STEAM education for students which helps them prepare for the world of work and can lead to amazing careers.

Now it is time for you and your students to begin a very excellent adventure in robotics.





# LESSON 1: Motor Control – Line Tracker



Robotics technology plays a very important role in almost every aspect of today's life from small toy vehicles to drivable toys, electric cars and even to rockets that carry people and satellites into space. The K'NEX Education Robotics Set will introduce students to their first K'NEX hands-on robotics experience. Students will build model robotic systems that they program to move and respond to information gathered from the environment. This first lesson will introduce students to the concepts related to motors and their potential when used in robotic systems. This is a project-based learning opportunity that allows students working in collaborative groups to explore, investigate, and experiment. Students will follow instructions provided in the K'NEX Education Robotics Instruction Booklet to build the Line Tracker model that they will use during this lesson.

## Objectives

*Students will be able to:*

- Work in collaborative teams.
- Build models from instructions in the Robotics Instruction Booklet.
- Design and write programs using flowcharts to control their model.
- Design techniques and/or experiments to test the performance of their model.
- Experience real world applications of STEAM concepts.

## Materials

*Each group will need:*

- Built Line Tracker model (*page 64 in the instructions book*)
- Metric tape and/or metre stick
- Graph paper
- Open area in the classroom or hallway for the model to operate
- K'NEX Robotics User's Guide
- PC and K'NEX Robotics Programming Software
- Student Instruction Sheet A - Lesson 1
- Copies of Student Response Sheets 1 & 2 - Lesson 1

## Teacher's Notes

- Allow time for students to explore the K'NEX pieces (*connectors, rods, wheels, etc.*) before they begin building the model (*students may need assistance building the Line Tracker model*).
- Allow time for students to explore the K'NEX Robotics Building System User's Guide before they begin using the K'NEX Robotics Programming Software.
- It is recommended to build the Line Tracker model first, with all motors and sensors connected to the Control Box, before trying to write programs. Line tracking is a classic robotics application, that integrates sensing, actuation, and control algorithms
- When the Control Box is connected (via USB cable) to the K'NEX Robotics Programming Software, the software should recognize all the sensors, motors, and the ports where they are connected into the Control Box.
- Before programming, have the students click on each motor icon in the software to operate the motors in a "simulation" mode. This will help them discover which motor icon is the "left" motor and which one is the "right" motor.

## Introductory Robotics Lesson

This lesson is designed for educators teaching an introductory robotics class. Students will measure the distance the Line Tracker travels when they change the speed of the motor using the K'NEX Robotics Programming Software.

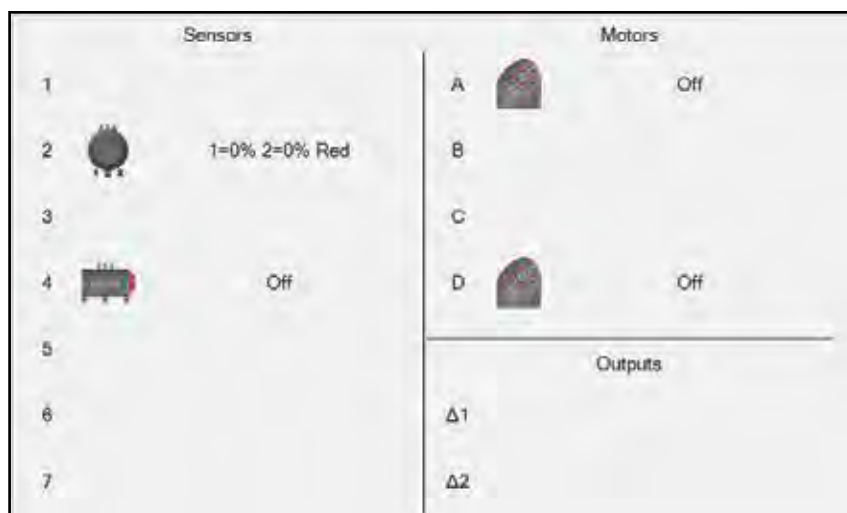
### Lesson 1 includes:

- Line Tracker Building Instructions – step-by-step plans from the Robotics Instruction Booklet.
- Student Instruction Sheet A to allow students to write their first program which will be used as they collect, present, and analyse data.
- Student Response Sheet 1. Students will keep records and results of their experiments on this page.
- Optional challenge activities provide additional programming opportunities to strengthen student understanding and for student groups who finish early. Student Response Sheet 2 is provided for students to keep records of their programming strategies.
- Clean up time.
- Optional end-of-class discussion activity (*at teacher's discretion*).

## Exploring the K'NEX Robotics Programming Software and Building the Line Tracker Model

### Pre-Lesson:

- Identify student groups of 2 - 3.
- Build the Line Tracker Model from the Robotics Instruction Booklet provided.
- Provide groups a computer with K'NEX Robotics Programming Software and K'NEX Robotics Building System User's Guide. The K'NEX Robotics Programming Software and the K'NEX Robotics Building System User's Guide can be downloaded on as many computers as needed for instruction.
- Instruct students to explore the User's Guide to familiarise themselves with the K'NEX Robotics Programming Software. *NOTE: Based on your student's experience level, you may need to provide some level of direct instruction to familiarize them with K'NEX Robotics Programming Software.*
- Specific motor and sensor port connections must be known to correctly program the model. After connecting the USB cable to the Control Box and clicking the "Connect" button in the top menu bar, you should see the icons pictured in Figure 1 in the I/O panel.



**Figure 1:** Light Sensor: Port 2, Push Button Switch: Port 4, Right Motor: Port D, Left Motor: Port A

- The motors in the Line Tracker model are positioned so that when the motors are programmed to run forward, the model will also move forward.
- Encourage students to design sample programs that include motor commands.

### Lesson Process:

- Provide students with copies of Student Instruction Sheet A. These instructions walk students through the process of writing and running their first program.
- Review the instructions to ensure that the students understand the information that is provided to make their programming experience more successful.
- Distribute Student Response Sheet 1 and review the directions along with the charting and graphing requirements of the activity. Also, introduce Student Response Sheet 2 if the students will continue on to complete Challenge Activities.
- Student Instruction Sheet A directs students as they write their first flowchart program to operate their model.
- The figures and text below provide you with solutions to the programs that the students are required to design as they complete Student Response Sheets 1 & 2.
- Provide metric tape/metre sticks and graph paper for students.

## Sample solutions to the programs students will need to complete as they work on both Student Response Sheets

Students will use their first program (*outlined on Student Instruction Sheet A*) as a template to develop the second, third, and fourth programs required to complete Student Response Sheet 1.

### The Flowchart sample solution for the students' second program

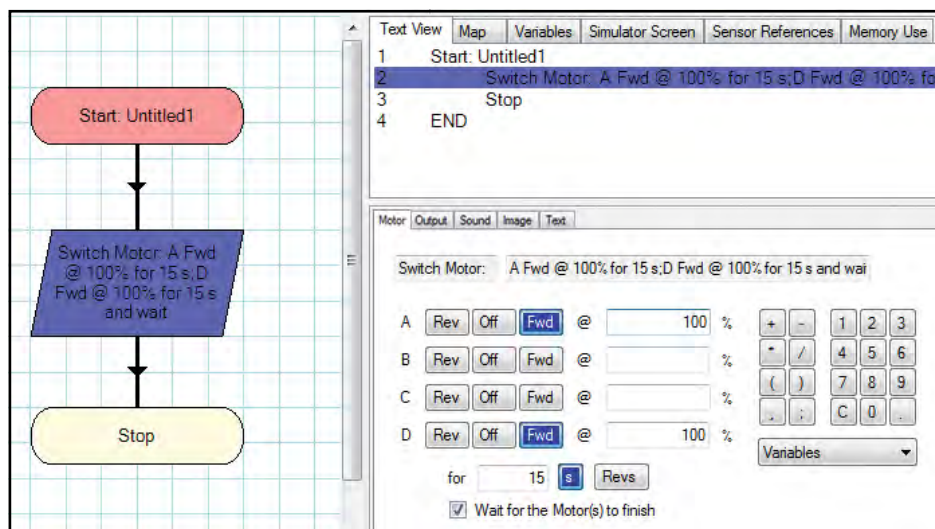


Figure 2: Line Tracker driving forward for 15 seconds with 50% power.

### The Flowchart sample solutions for the students' third and fourth programs

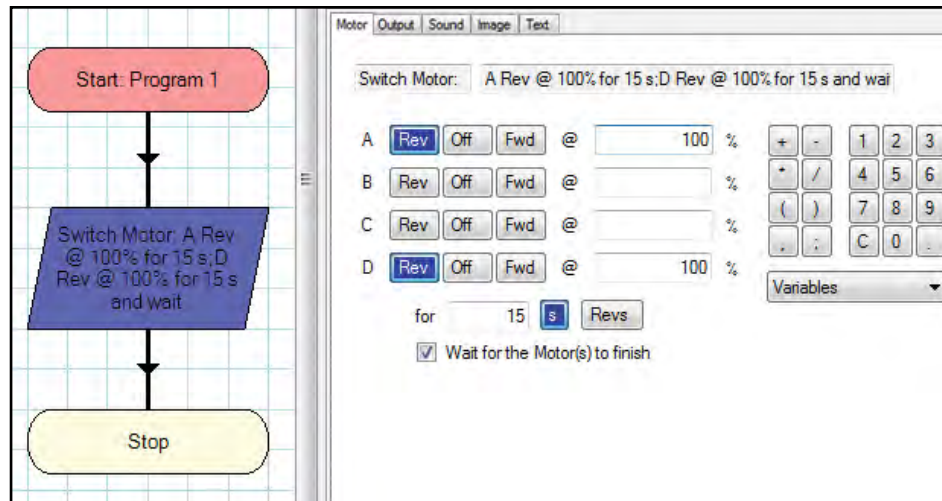
The flowcharts for the third and fourth programs will be identical to the flowchart in Figure 2 except the percentages will be 25% and 75% respectively.

## Solutions to Student Response Sheet 2 Challenges

### STUDENT CHALLENGE: Program the model to move backwards

**Teacher's Note:** Distribute Student Response Sheet 2 and remind students that they are to thought shower and draw the flowchart for their proposed programs before using the computer. Ask them to explain their logic orally as you move about the classroom.

The first challenge is to write and test a flowchart for the Line Tracker to move backwards. Once the flowchart is initiated, the students will set motor A to 'Rev' (*reverse*) and select a power level. They will also set motor D to 'Rev' (*reverse*) and select the same power level they chose for motor A. Students then must set the time duration, in seconds, for the motors (*Figure 3*).



**Figure 3:** Line Tracker moving backwards for 15 seconds with 100% power.



**STUDENT CHALLENGE: Turn the Line Tracker to the right**

**Teacher's Note:** Have the students explore different motor controls to make a 90° right turn.

- **Arc Turn Right** – Both motors turn the same direction, but the right motor is powered less than the left. This arrangement will cause the model to turn in an arc to the right. Challenge students to explore what happens to the size of the turn when larger or smaller power differences are used. *NOTE: Adjustments to the duration (in seconds) need to be made to keep the right turn approximately 90°, and not have the model turn 180° or spin in a circle. Figure 4 shows an example of the Arc Turn Right.*

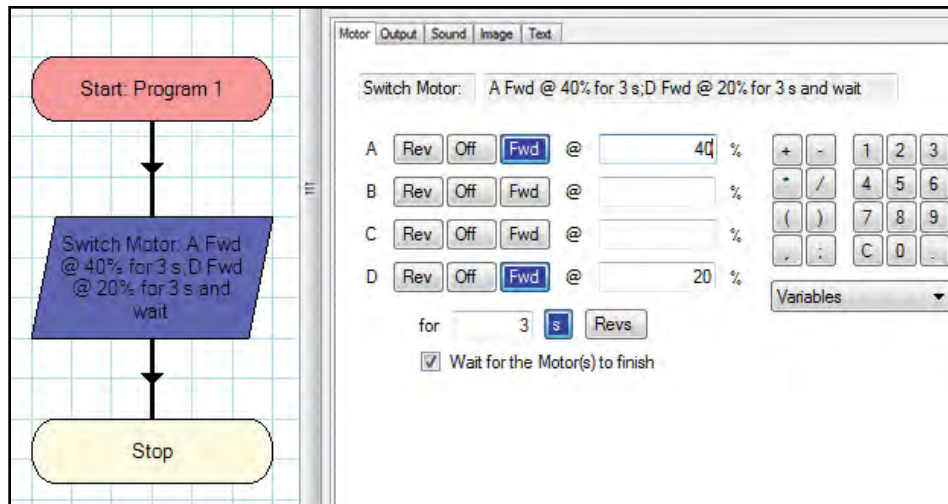


Figure 4: Arc Turn Right.

- **Pivot Turn Right** – Right motor spins in the opposite direction to the left motor causing the model to pivot in place as it turns to the right (See Figure 5).

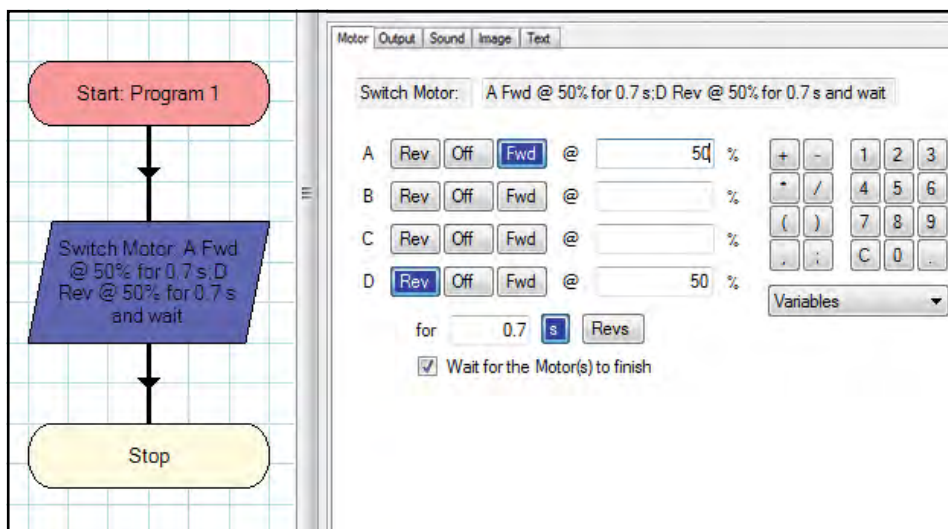


Figure 5: Pivot Turn Right.

**STUDENT CHALLENGE: Turn the Line Tracker to the left**

**Teacher's Note:** To program the model to turn left, students will use information similar to the data in Figures 4 & 5 except they will reverse the motor directions and power settings for motors A and D.

## LESSON 1 – Student Instruction Sheet A

### Programming and Investigating the Motion of the Line Tracker

- Gather your model, USB cable, computer, a metre stick or metric tape, and a sheet of graph paper.
- After receiving your teacher’s approval, start the programming process to meet the requirements of the activity on Student Response Sheet 1.
- You will be using two motors, and the K’NEX Robotics Control Box to operate the Line Tracker.

### Introduction to Motors

Motors are used in almost all robotic projects. Anything that moves has a motor of some form. The two motors used will control the motion of the Line Tracker. The motors need power that is provided through the K’NEX Robotics Control Box. The motor ports (A, B, C, D) are on the top side of the Control Box. Letter names for each motor port are on the front face above the LCD screen – see Figure 6.



Figure 6: K’NEX Robotics Control Box - Front Face

### Connecting the Control Box to the Programming Software

After the Line Tracker is built, ensure that the Control Box and the two motors are attached and inserted in the model as per the Line Tracker instructions. Launch the K’NEX Robotics Programming Software and select “USB connection”. Turn the power switch on the Control Box to “ON.” Attach the USB cable between the Control Box and your computer. Find and click on the ‘Connect’ icon in the control toolbar at the top of the screen. Specific motor and sensor port locations must be known to correctly program the model. When the software is connected to the Control Box of an assembled model, it will automatically populate the motor and sensor ports that are connected. After connecting the USB cable to the Control Box and clicking the “Connect” button on the top menu bar, you should see the icons pictured in Figure 7. Note that the Colour/Light Sensor and Push Button Switch are on the model and visible in the software screen, but will not be used until Lesson 2. Refer to the troubleshooting guide if you are having difficulty making a connection between the box and your computer.



Figure 7: Software screen with correct control box connections.

## PROGRAMMING

### First Program: Moving the Line Tracker Forward

To write your first flowchart program, select the required icons one at a time from the symbol toolbar on the left of the screen and place them in the edit area. Select and place an “Output” and a “Stop” icon as shown in Figure 8.

The start icon is provided by default on every new workspace for programming. You may choose to name your program by clicking on the Start icon and typing a name for the program, i.e., “Forward”, in the space that opens at the bottom right of the screen. The name will appear as “Start: Forward” in the start oval once you type the name and press enter on the keyboard or click the “OK” button on the screen.

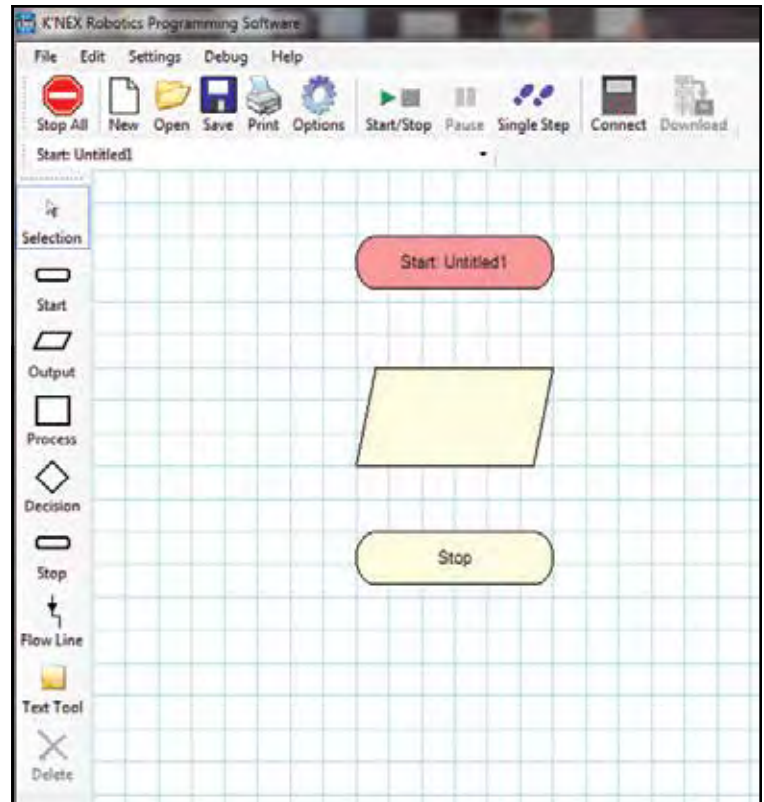


Figure 8: Added “Output” and “Stop” program icons.

Now, connect the blocks using the flow line icon. First, select the “Flow Line” icon from the symbol toolbar. Second, click on the “Start” oval in the edit area, and then click on the “Output” symbol just below it. The “Start” and “Output” program items should now be connected with a flow line. Repeat the process to connect the “Output” with “Stop.” If your flowchart looks like Figure 9, you are ready for the next step.

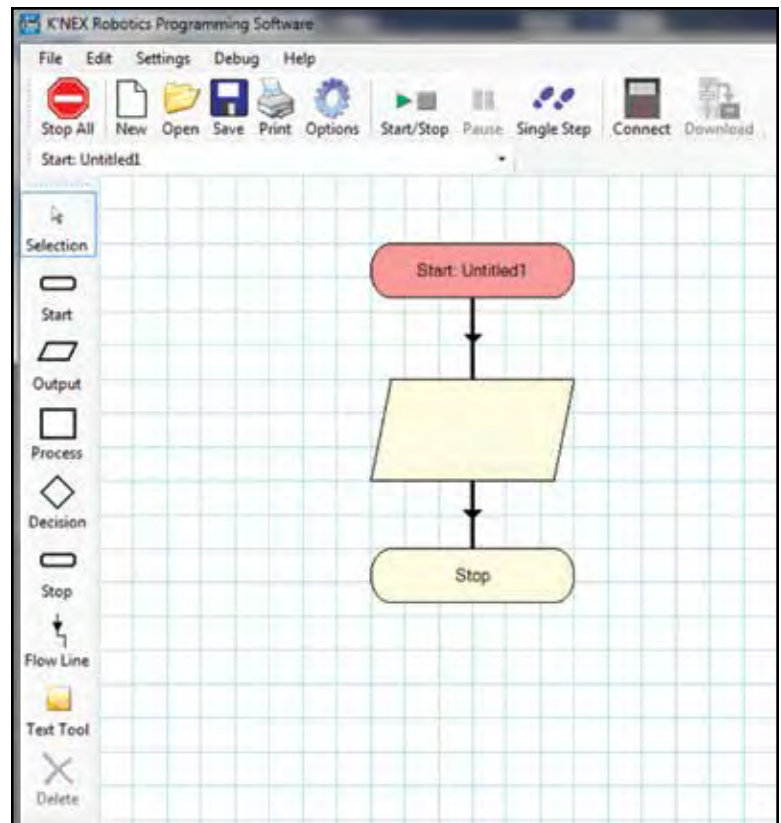


Figure 9: Added flow lines to program.



Click the selection tool from the symbol toolbar and select the “Output” symbol (*parallelogram shaped box*) in your program. The shape will highlight and the lower right window (*Input/Output panel*) will change. You will see six tabs in this new window. Click on the “motor” tab and motor programming options will appear, see Figure 10.

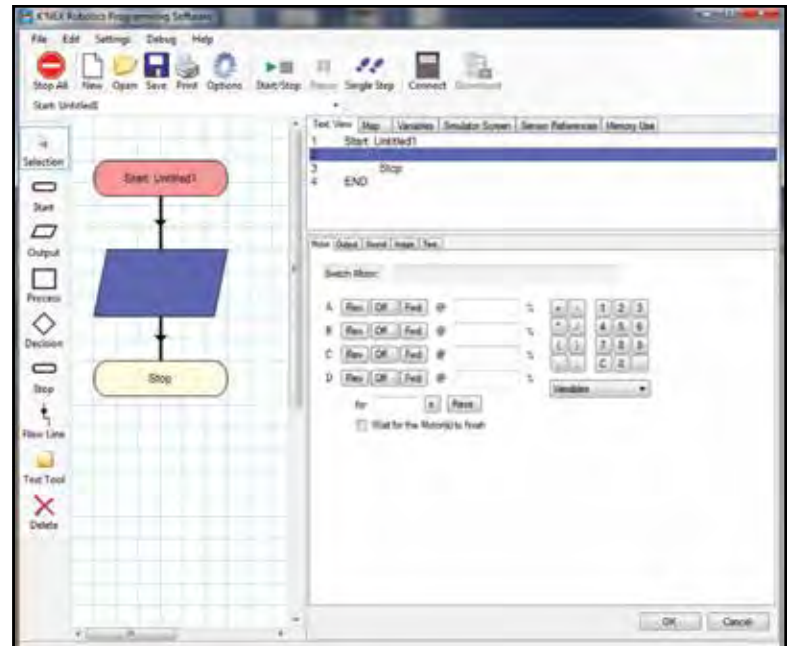


Figure 10: Switch Motor Program Option

To program the motors correctly you need to know the position of the motors in the Line Tracker model and the ports the motors are using on the Control Box. For this model the right side motor is plugged into port D and the left side motor is plugged into port A. You will need to program the motors to run forward in your first program. To answer the question, “How far does the Line Tracker travel when you program the model to move forward at 100% motor power for 15 seconds,” you will need to enter motor parameters as shown in Figure 11. Be sure to select the “wait for motor to finish” box so that the motor runs for 15 seconds. Otherwise the motor will only run for a fraction of a second before the program comes to a stop. You also need to click “OK” to save your entered motor parameters.

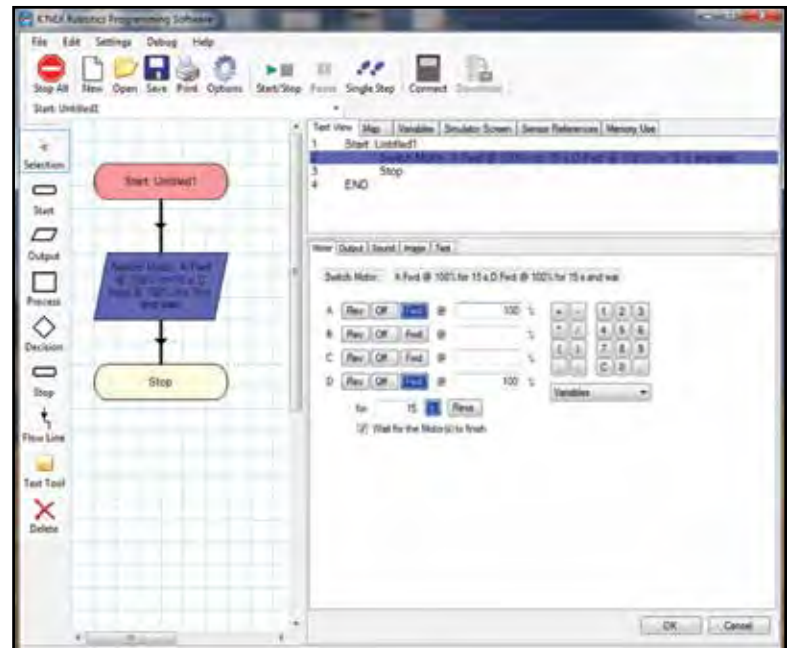


Figure 11: Line Tracker Program – moving forward for 15 seconds with 100% power

If your program looks like Figure 11, then you are ready to download it to the K'NEX Robotics Control Box. Click the “Download” icon on the control toolbar near the top of the screen. In the pop-up window, select a location for your program by clicking one of the numbered rows (*the selected space will turn blue*). Click “Download Program” and a “Control Program Name” pop-up window will appear. Enter the name of your program in the space provided and click the “OK” button. Your program has been downloaded to the Control Box. The program name will be shown in parentheses after the file name at the top of the program window. Click the red “X” on the pop-up window to exit. You can now run the program from the computer using the “Start/Stop” button from the control toolbar. To run the program remotely from the Control Box, click the “Disconnect” button from the control toolbar and unplug the USB cable. Select the program you downloaded from the menu screens on the Control Box, set the model at the starting point and run the program to see how far the model travels in 15 seconds. At this point you have a program to help you complete Student Response Sheet 1.

# Student Response Sheet 1

Name \_\_\_\_\_ Group # \_\_\_\_\_

How far does the Line Tracker travel when you program the model to move forward at 100% motor power for 15 seconds? How far does the model move forward at 50% motor power for 15 seconds?

Distance traveled at 100% motor power in 15 seconds: \_\_\_\_\_

Distance traveled at 50% motor power in 15 seconds: \_\_\_\_\_

Briefly describe what this information tells you about the performance of the Line Tracker model.

Insert the results from the previous activity into the chart below and write additional programs to collect enough data to complete the chart.

Motor Power	Distance Travelled	Comments
25%		
50%		
75%		
100%		

Identify the independent variable shown on your data chart.

Identify the dependent variable.

Graph your results. *(Attach your graph to this page)*

Analyse your graph and explain what it shows about the relationship between motor power and distance travelled by the model.



# Student Response Sheet 2

Name \_\_\_\_\_ Group # \_\_\_\_\_

Diagram a flowchart below, that will allow the Line Tracker to move backwards for 15 seconds.

Program the model to move backwards and test its operation.  
Did the Line Tracker move as you expected? Explain!

**LESSON 1**  
**Student Response Sheet 2**  
**(continued)**

Name \_\_\_\_\_

Group # \_\_\_\_\_

Diagram a flowchart below, that turns the Line Tracker to the right.

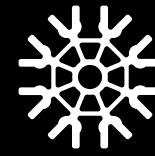
Diagram a second flowchart to turn left.

Program the Line Tracker to complete the turns and test your programs.

Did the Line Tracker turn in each case as you expected? Explain!



## LESSON 2: Line Tracker



The Line Tracker will be the first interactive robot students build that requires motors and a sensor to work together. In the previous lesson (*Motor Control – Line Tracker*) we introduced the concept of the motor, its performance and motion (*forward and reverse*). In Lesson 2, we will add another feature where the Line Tracker moves using motors and has a sensor that acts like an eye to check where it is going. The Line Tracker will have a feedback sensory system that interacts with the motors.

The Colour/Light Sensor included in this feedback sensory system helps to determine (*or continuously check*) if the sensor is above a white or black area. In simple terms, the sensor (*The K'NEX sensor is actually made up of a single unit that includes two small sensors mounted next to each other*) acts like our eye and sends signals back to the Control Box which then sends commands to the Line Tracker to go forward or turn left or right.

### Objectives

*Students will be able to:*

- Work in collaborative teams.
- Build the Line Tracker model or use the model from Lesson 1.
- Design and write a main program that includes procedures to enable the Line Tracker to follow a black line around a track drawn on the K'NEX Colour Mat.
- Experience and practice using a feedback sensory system.

### Materials

*Each group will need:*

- Built Line Tracker model (*Page 64 in the instructions book*)
- K'NEX Colour Mat for Colour/Light Sensor activities
- Open area in the classroom or hallway for the model to operate
- PC and K'NEX Robotics Programming Software
- K'NEX Robotics User's Guide
- Student Instruction Sheet A – Lesson 2
- Copies of Student Response Sheets 1, 2, & 3 - Lesson 2

### Teacher's Notes

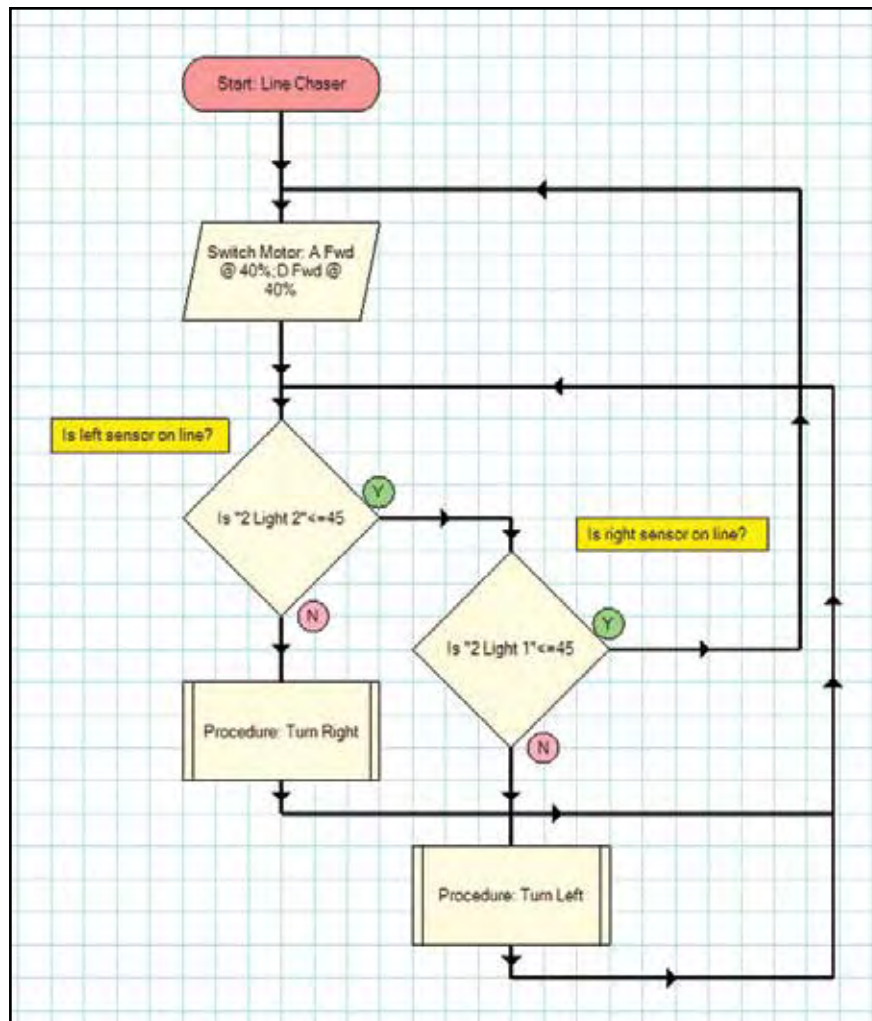
- Identify student groups of 2 - 3.
- Allow the students to explore the K'NEX Education materials and give them time to review the K'NEX Robotics User's Guide.
- Require that students show their progress before they begin to program their model.
- The Line Tracker model should be programmed to operate at low motor speeds so that there is enough time for the microcontroller to process the information and direct the motors to operate the Line Tracker smoothly. Once the robot follows a line successfully, ask the students to increase the speed of the Line Tracker to explore the limits of the system. Students may also need to experiment with the light values used to optimise performance.
- It is recommended that the supplied transformer be used to power the models while writing programs to conserve battery power for remote use. The Control Box will only draw power from the transformer when it is connected; when the transformer is disconnected from the Control Box it reverts immediately to battery power. The transformer will not charge batteries installed in the battery compartment.

**The task for students is to:**

- Explore the logic flow of a program that can power the Line Tracker to follow a black line.
- Program the model to actually follow a black line on the K'NEX Colour Mat.
- Troubleshoot and redesign the program to ensure the model moves smoothly in their classroom environment.
- Complete one or more challenge activities to improve the operation of their model.

**Explore a Program that can power the Line Tracker along a Black Line**

Provide students with a copy of the flowchart shown in Figure 1.



**Figure 1:** Line Tracker Main Program Flowchart

Ask students to review and explore the logic used in the flowchart. The K'NEX Robotics User's Guide will enable them to explore the various Decision and Output symbols used in the program. Ask them to explain in a written paragraph or list format how the program will operate to ensure that the Line Tracker actually stays on the line. Review the responses in a large or small group setting to ensure that students are able to follow the logic displayed by the flowchart.

As shown in the main program flowchart, the students will produce separate procedures that will power the Line Tracker along a black line on the K'NEX Colour Mat. The directions to complete the program are provided on Student Instruction Sheet A – Lesson 2.

## LESSON 2 – Student Instruction Sheet A

The following activities will introduce you to the use of sensors in robotic systems and will allow you to program the Line Tracker robot to respond to sensor input.

### Requirements:

Complete Student Response Sheets 1, 2, and 3 as you work on this activity to keep a record of your work.

### Introduction to Sensors

Sensors are an important part of any robotic design as they act like a feedback system. They provide information to the microcontroller, which acts like our brain to make decisions and send information to the motors that power the model. In this lesson, you will use the Colour/Light Sensor to measure digital light values. Before programming, the operation of the Colour/Light Sensor should be evaluated to determine the digital values the sensor identifies for black and white areas of the K'NEX Colour Mat.

### Before you begin:

- Review the building instructions for the Line Tracker.
- Read and review the K'NEX Robotics Programming Software User's Guide as it pertains to Decision and Output programming symbols.
- Read the Colour/Light Sensor description provided in the K'NEX Robotics Programming Software User's Guide.

### Steps:

1. Build the Line Tracker. Add the Control Box, Colour/Light Sensor, Push Button Sensor and motors as shown in the Building Instructions.
2. **DO NOT** forget to power the Control Box. K'NEX recommends using the supplied transformer if possible during programming to conserve battery power.
3. Open the K'NEX Robotics Programming Software.
4. Click the 'connect' icon on the K'NEX Robotics Programming software to connect the software with the Control Box.
5. You will see Colour/Light Sensor readings on line 2 under the "Sensors" heading in the I/O panel on the lower right side of the screen. Line 2 corresponds to Port 2 on the Control Box where the sensor is connected.
6. Place the model so that the Colour/Light Sensor is on a white section of the mat.
7. Collect sensor readings at three white areas on the mat, enter them on Student Response Sheet 1 and average the data.
8. Place the model so that the Colour/Light Sensor is on the black line on the mat.
9. Collect sensor readings at three black areas on the mat, enter them on Student Response Sheet 1 and average the data.
10. The average data values calculated on Student Response Sheet 1 will help decide a value that can be used to separate readings that indicate the sensor is above a white area or a black area.
11. After our testing at K'NEX, we determined that a sensor reading less than or equal to 45 indicates black and any value greater than 45 indicates white. Your values may vary from these results, please select your value as per the readings you receive from the sensor.
12. Once you have the digital values for the sensor, turn the model off, set it aside and start the logic design activity on Student Response Sheet 2.
13. Move on to the programming section for further instructions.



## PROGRAMMING

### Program Logic:

- The Line Tracker needs to know how to turn left, turn right, or go forward if it is to successfully follow the black line. Create separate procedures for each direction; LEFT and RIGHT.
- The Colour/Light Sensor is actually made up of two smaller sensors mounted a small distance apart inside a single housing. These smaller sensors are labeled “light 1” and “light 2” in the software. As positioned in the Line Tracker model, sensor “light 1” is on the right side and sensor “light 2” is on the left side.
- For this lesson the goal is to follow the black line by keeping both sensors over the line.
- Based on our sensor readings at K’NEX, if sensor “light 1” has a reading of 60 and sensor “light 2” has a reading of 25, the Line Tracker is partially off the right side of the black line. The program will have to turn the model to the left to return the Line Tracker to the line.
- If sensor “light 1” has a reading of 25 and sensor “light 2” has a reading of 60 the model is partially off the left side of the black line. The Line Tracker will have to turn right to return to the black line.

### How to create a procedure?

You should see a Start oval at the top of the edit area. Click on the Start and enter the procedure name “Turn Right” in the data box provided in the lower right portion of the screen. You can toggle between a “Start” and “Procedure” by clicking the labelled tabs just above the data box. Click the procedure tab and click OK. “Procedure: Turn Right” will appear in the oval and you are now ready to create the Turn Right procedure. *NOTE: Start ovals will be red, Procedure ovals will be blue.*

### PROCEDURE: “Turn Right”

The Line Tracker must turn slowly in order to avoid overshooting the black line. The procedure logic for “Turn Right” is shown in Figure 2. The example suggests switching motor A forward at 15% power while at the same time motor D is powered in reverse at 30% power for 0.15 seconds. The “wait for the motor(s) to finish” box is checked so the motors run for the time specified. To see if the model has been rotated back onto the line, a Decision symbol is added to check the input from the Colour/Light Sensor. The statement in the Decision symbol asks “Is 2 Light 2 <= 45”. This tells the Control Box to check the value of Sensor “light 2”, (the left side) of the Colour/Light Sensor plugged into port 2. If the sensor value is greater than 45 the sensor is still off the left side of the line. The model has not turned far enough and the right turn will be repeated. If the sensor value is less than or equal to 45, (black) the model is back on the line and the procedure exits back to the main program. *NOTE: It is best to always add a Stop to the end of each of your procedures.*

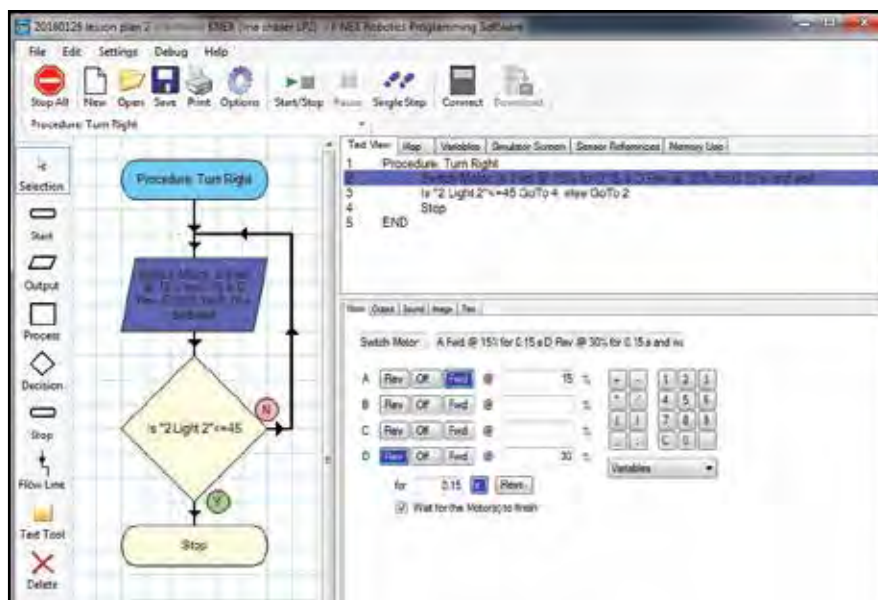


Figure 2: Turn Right.

**PROCEDURE: “Turn Left”**

If you successfully followed the logic to write the “Turn Right” procedure, you should have no problem programming the “Turn Left” procedure. Click on the Start icon from the symbol toolbar to get a new Start oval for the Turn Left procedure. You will notice immediately that the “Turn Right” procedure has “disappeared” into the background. You can toggle between procedures and starts by selecting the desired option from the pull down menu below the control toolbar. You can also click on different procedures and starts from the map view in the upper right side of the screen. If the map tab is selected, click on the desired procedure or start to activate it. Create the procedure for a left turn. Refer to Figure 3 if necessary.

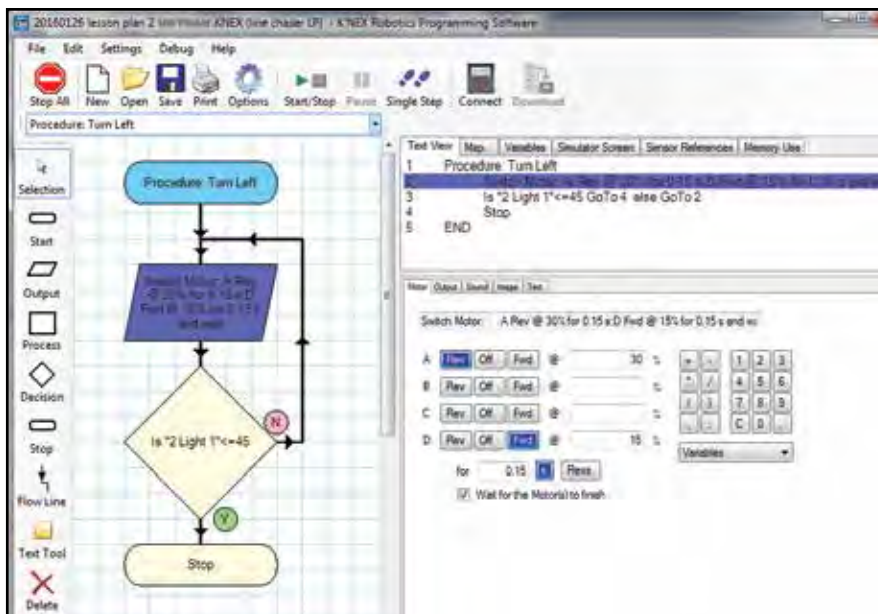


Figure 3: Turn Left.

**Main Program:**

Click the “Start” icon from the Symbol toolbar on the left of the screen to start a new flowchart for the main program. Click on the “Start” oval in the edit area and enter an appropriate name for the main program. Be sure to select the Start Tab rather than the Procedure Tab when entering a name for the program.

When you operate the Line Tracker on the K’NEX Colour Mat, you will start the model with the Colour/Light Sensor centered on the black line. This means the first step in your program should be to make the model move forward. Place an Output Symbol under the Start in the edit area.

Based on our experience at K’NEX, a good starting speed for the Line Tracker is 40% motor power.

*NOTE: See Figure 4 Main Program, if you are unsure of how to complete this task.*

You are now ready to complete your program using the procedures you prepared earlier. The program will need to constantly check sensors 1 & 2 to be sure the model remains on the black line. To make that happen, you will need to add some decision symbols and loops to your program. The decision symbols will guide the program based on the sensor readings and the loops will allow the program to check the sensors over and over again.

A copy of the Line Tracker program developed at K’NEX is provided in Figure 4. The program has been provided to allow you to get your model operational. Test your model, troubleshoot its performance, and make changes to your procedures and program as necessary to improve the model’s operation. Record any changes you make and the reason for those changes on Student Response Sheet 2 – Lesson 2.

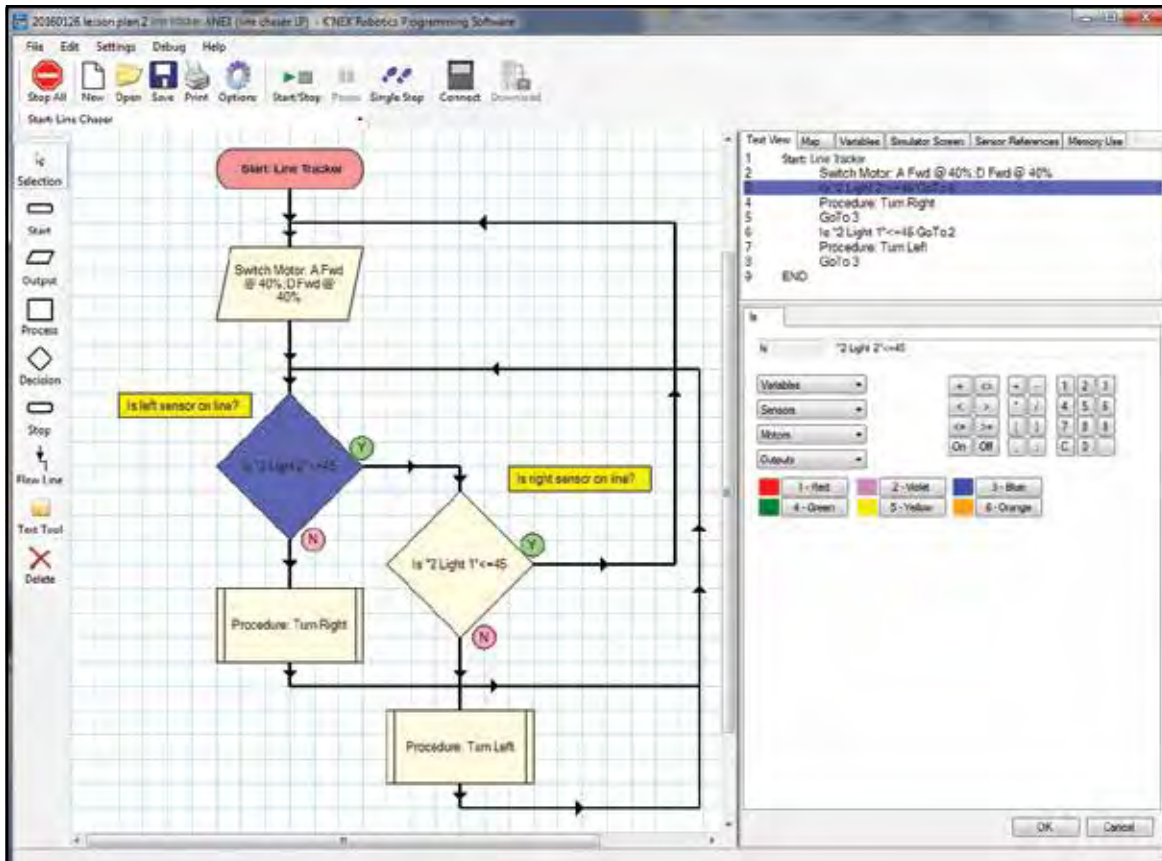


Figure 4: Main Program.

**Challenge:** Refer to Student Response Sheet 3 – Lesson 2 for a challenge activity for your group to complete.

# Student Response Sheet 1

Name \_\_\_\_\_ Group # \_\_\_\_\_

Measure the Colour/Light Sensor digital readings for the black portion of the mat.

Colour/Light Sensor Readings	Sensor 1 – Black Colour	Sensor 2 – Black Colour
Trial 1		
Trial 2		
Trial 3		
Final Average		

Explain your results and describe how this information will help as you program the Line Tracker to follow the black line.

Calculate the Colour/Light sensor digital readings for the white portion of the mat.

Colour/Light Sensor Readings	Sensor 1 – White Colour	Sensor 2 – White Colour
Trial 1		
Trial 2		
Trial 3		
Final Average		

Explain your results and describe how this information will help as you program the Line Tracker to follow the black line.

# Student Response Sheet 2

Name \_\_\_\_\_ Group # \_\_\_\_\_

Describe any changes that you made to the initial program provided by K'NEX to make the Line Tracker operate more efficiently in your classroom or setting.

What did you notice about the performance of the Line Tracker that led you to decide to make adjustments to the program?

List any changes you made to the program? Describe the logic behind those changes.

Draw or attach copies of the flowcharts for the changes you made to the program.



# Student Response Sheet 3

Name \_\_\_\_\_

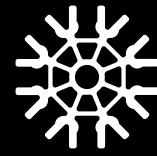
Group # \_\_\_\_\_

**Challenge** – In the Main program shown in Figure 4, flow lines that emanate from the Turn Right and Turn Left procedures loop back to the first Decision symbol. In contrast, the flow line from the second Decision symbol loops back to the Output symbol.

Discuss the logic involved in this arrangement with your group and explain why this strategy is necessary in writing below.



## LESSON 3: Bi-Ped Robot



The Bi-Ped Robot model used in this lesson will help you program using strategies that are similar to how humans process information for simple tasks. In previous lessons (Motor Control and Line Tracking with the Line Tracker model), the interaction of sensors and motors were explored and practiced. In Lesson 3, the feedback system and communication between the Control Box, motors and sensors are used in more advanced ways to program the Bi-Ped Robot. The Robot has three sensors: Distance, Colour/Light and Push Button. It also has four motors: one motor to rotate the head, two motors to power the legs and one motor to open/close the hand to grab or drop an object.

The Distance Sensor can measure the distance the robot is from an object, the Colour/Light Sensor will determine if something of a certain colour (*yellow*) is in the robot's hand, the Push Button is used as the operator determines when to grab an object (*ball*) that is not yellow.

The programming structure in this lesson will introduce a method called "procedure inside a procedure". This modular programming structure will expose students to a real time programming experience where even a simple task like grabbing a ball needs a structural programming flow to operate correctly and smoothly. This advanced level of logic is critical to student learning at an early age so they can investigate and practise modularising the codes/procedures they use to optimise other programming strategies.

### Objectives

*Students will be able to:*

- Work in collaborative teams.
- Design and write programs including embedded procedures.
- Create multi-layer procedures (*procedure inside a procedure*).
- Experience real world applications of STEAM concepts.
- Develop a logic strategy and convert that logic into a structured program.

### Materials

*Each group will need:*

- Bi-Ped Robot model (*page 27 in the instructions book*)
- Open area in the classroom or hallway for the robot to walk
- PC and K'NEX Robotics Programming Software
- K'NEX Robotics User's Guide
- Copies of Student Instruction Sheet A - Lesson 3
- Copies of Student Response Sheets 1, 2, & 3 - Lesson 3

### Teacher's Notes

- Identify student groups of 2 - 3.
- Allow students to explore the K'NEX Education materials and give them time to review the K'NEX Robotics User's Guide.
- Students may need help building the robot.
- Provide students with a screen shot that shows the completed main program.
- Complete the structured portion of the lesson with students.

- Request that students describe their thought processes and model building progress before they begin further programming with their model.
- Students may need help as they program.
- Ask students to verify the direction the motors turn so that the robot will walk forward.

The students will learn to write separate procedures within procedures that they can incorporate into a main program in order to power the Bi-Ped Robot to look around, move forward, grab a ball, and finally drop that ball. The structured part of this lesson introduces the main program and allows you to guide students as they write the beginning steps of their programs. You can then turn the programming responsibility over to the students beginning with the first procedure within a procedure as outlined on Student Instruction Sheet A – Lesson 3.

The following few pages include instructions for the procedures and process symbols that make up the structured portion of the lesson along with screen shots you can use as you assist the students as they begin their programming. For more detail on how to generate procedures, review the K'NEX Robotics User's Guide. If you would like students to take responsibility for programming the model independently, copy the screen shots on the next few pages to direct their efforts. They can then move on to Student Instruction Sheet A – Lesson 3 to complete their programs.

### Main Program:

The main program shows the logic that could be used to program the Bi-Ped Robot to grab a ball (Figure 1). A step-by-step method is described as follows:

1. The first block (*procedure*) in the flowchart as shown below in Figure 1 is "TURNHEAD." The procedure "TURNHEAD" instructs the Robot to move its head once left, once right, and back to straight ahead. The logic flow of the procedure "TURNHEAD" can be seen in Figure 2.

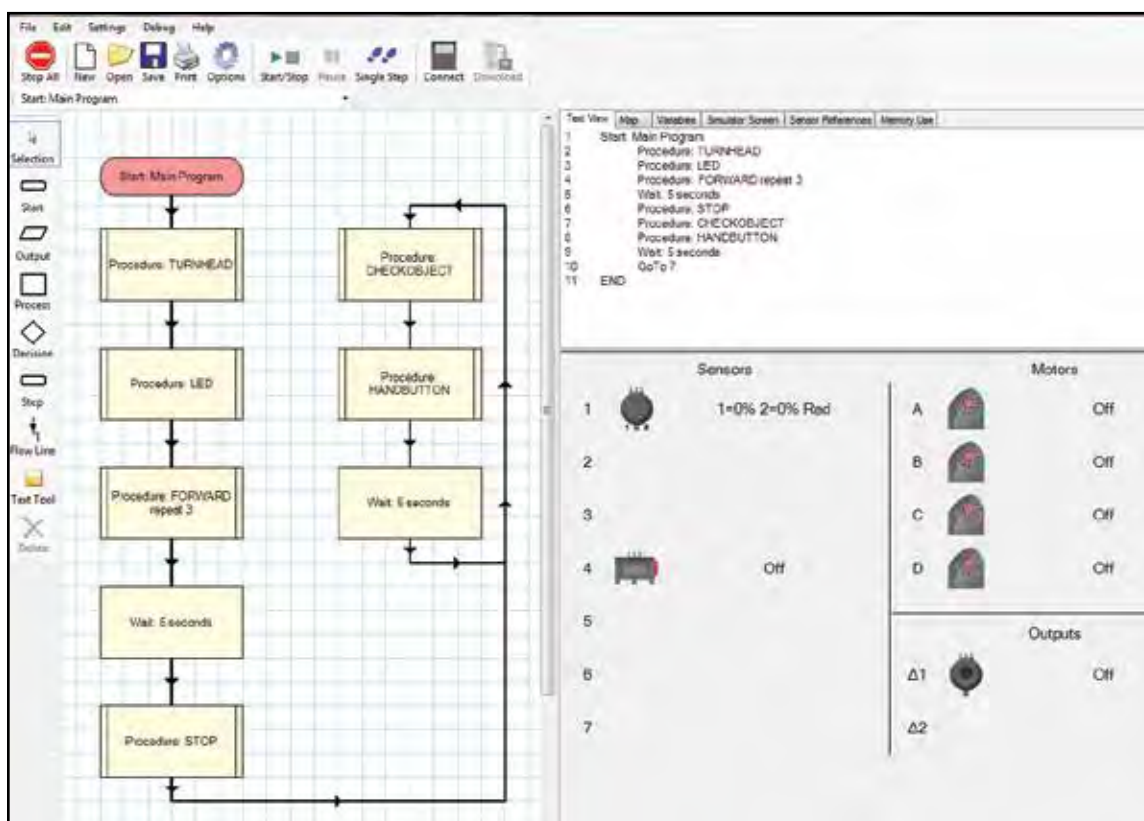


Figure 1: Main Program.

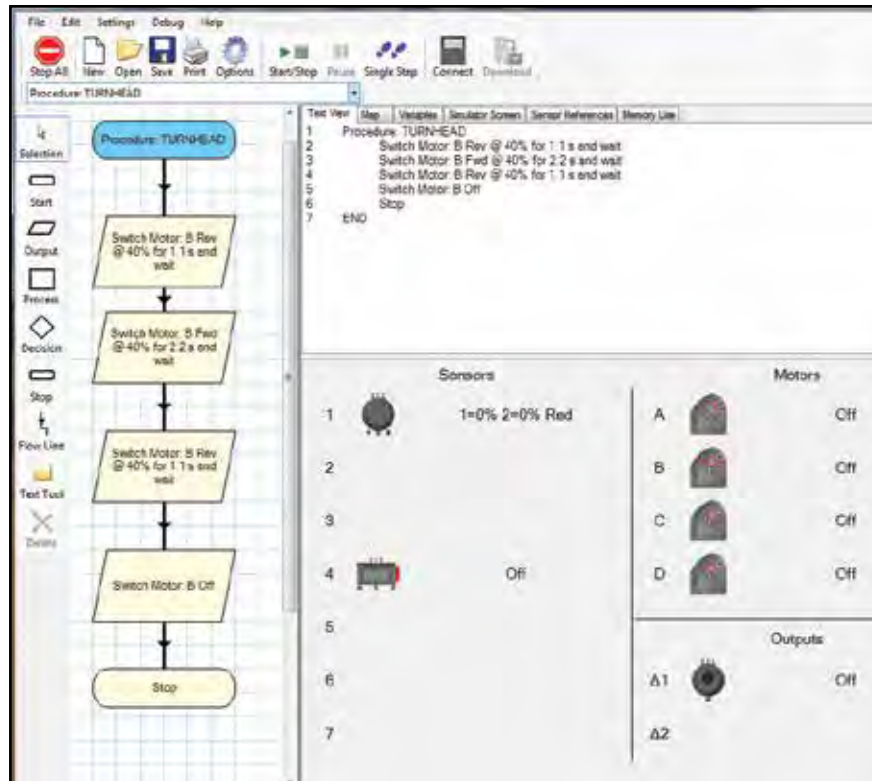


Figure 2: "TURNHEAD" Procedure.

To complete the "TURNHEAD" program as shown in Figure 2 complete the following steps.

- To turn the Robot's head, motor B turns at 40% speed in reverse (*that command turns the head to left*) for 1.1 seconds. Be sure to click the "Wait for motor(s) to finish" box when you set the motor percentage and the time for the motor to run.
- Next, motor B turns forward at 40% power for 2.2 seconds and directs the program to wait until the motor is finished (*the head turns right until the Robot is facing right*).
- The final command turns motor B in reverse at 40% for another 1.1 seconds to bring the head back to its original position.
- **NOTE:** *The motor speed needs to be slow enough for an observer to see the Robot turn its head as smoothly as possible. Test different motor speeds and time delays until your Robot moves its head smoothly. Do not forget to turn motor B off at the end of the procedure.*

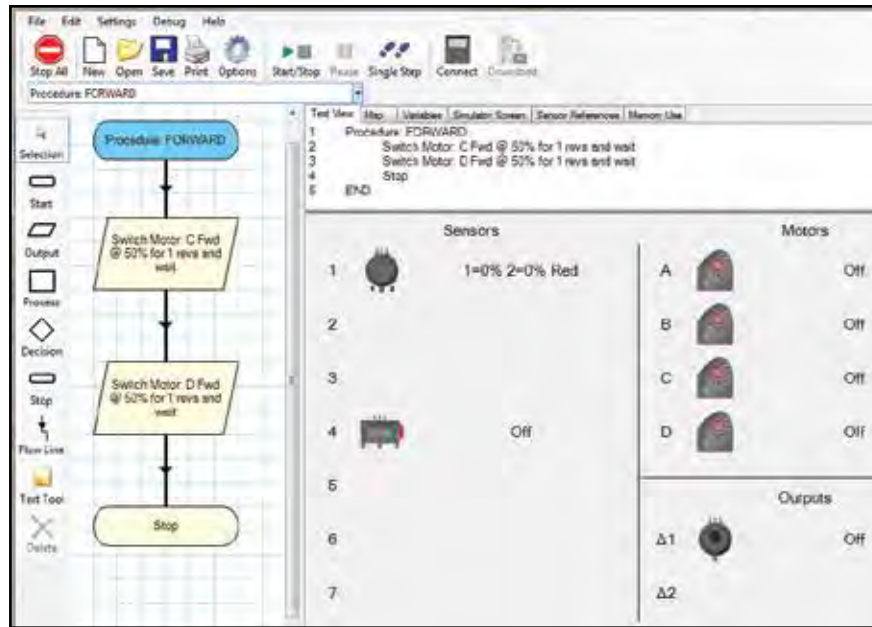


Figure 3: "FORWARD" Procedure.

The "FORWARD" procedure enables the robot to walk forward for several steps.

- From Figure 1 you can see that the program logic flows from the "TURNHEAD" procedure to the "FORWARD" procedure. Motors C and D are connected to the two legs of the Robot. Figure 3 shows the flowchart for the Robot walking. The motors (C and D) are programmed to move forward at 45% speed. As you follow the logic in "FORWARD" you will notice motor C completes a revolution and then motor D completes a revolution. This process is repeated three times (*When you designate the process symbol for "FORWARD" as a procedure you will name it "Forward" and place a "3" in the 'repeat' space to the right of the title*). The main program as seen in Figure 1 makes the Robot stop after a total of three repeats of the procedure. If you want the Robot to walk for more time, increase the number of repeats within the procedure.

To continue to the procedure inside a procedure, refer to Student Instruction Sheet A - Lesson 3. Steps 1 – 3 addressed the first section of the main program. The directions on the Student Instruction Sheets will allow students to develop the logic to complete the main program.



## Student Instruction Sheet A – Lesson 3

The following activities will introduce you to the use of the Push Button and Colour/Light Sensor in a robotic system and will allow you to program the Bi-Ped Robot to respond to sensor inputs. The sensor inputs will result in signals being sent to the motors to perform tasks. The main task assigned to the Bi-Ped Robot is to grab a ball and let go after a couple of seconds. Challenge activities are also included for you to attempt.

### Before you begin:

- Review the building instructions for the Bi-Ped Robot.
- Read and review the K'NEX Robotics Programming Software User's Guide as it refers to programs and procedures.

### Steps:

1. Build the Bi-Ped Robot per the K'NEX Building Instructions.
2. **DO NOT** forget to power the Control Box and to switch it on.
3. Open the K'NEX Robotics Programming Software.
4. Click the 'connect' icon on the K'NEX Robotics Programming Software.
5. Move on to the programming section below.

### PROGRAMMING

Previously you began a program to direct the Robot to move forward, grab a yellow ball and release it. The following steps will enable you to complete that program and provide some challenges to test your programming skills.

#### Grab and Throw Logic

The simple task of holding a ball and throwing is a process that requires a series of multiple signals in our brains. Our brain sends signals to our hands when the ball comes close to us to grab and throw it back. Isn't this how we play catch? Keep this in mind as you develop a logical sequence of actions to train and program the Robot to complete a similar action.

You will use an 'if - then' conditional statement as you program. For example, 'if' the ball is close, 'then' grab it. The same logic will enable you to create a procedure inside a procedure.

#### Procedure Inside A Procedure

When we want to hold something and drop it repeatedly, our brain processes these repetitive tasks and performs them quickly. You will program procedures within procedures to allow the Robot to complete repetitive tasks.

Program the Bi-Ped Robot to:

- Check the colour of a ball placed in its hand with the Colour/Light Sensor.
- "If" a yellow ball has been placed in its hand, "then" the hand will close and the Robot will take a few steps forward before dropping the ball.
- "If" the ball is not yellow, "then" the program will check to see whether or not the "Push Button" is pressed.
- "If" the "Push Button" is pressed, "then" the Robot will grab the ball for a short time before releasing it.
- "If" the "Push Button" is not pressed, "then" the program will repeat these steps over and over.

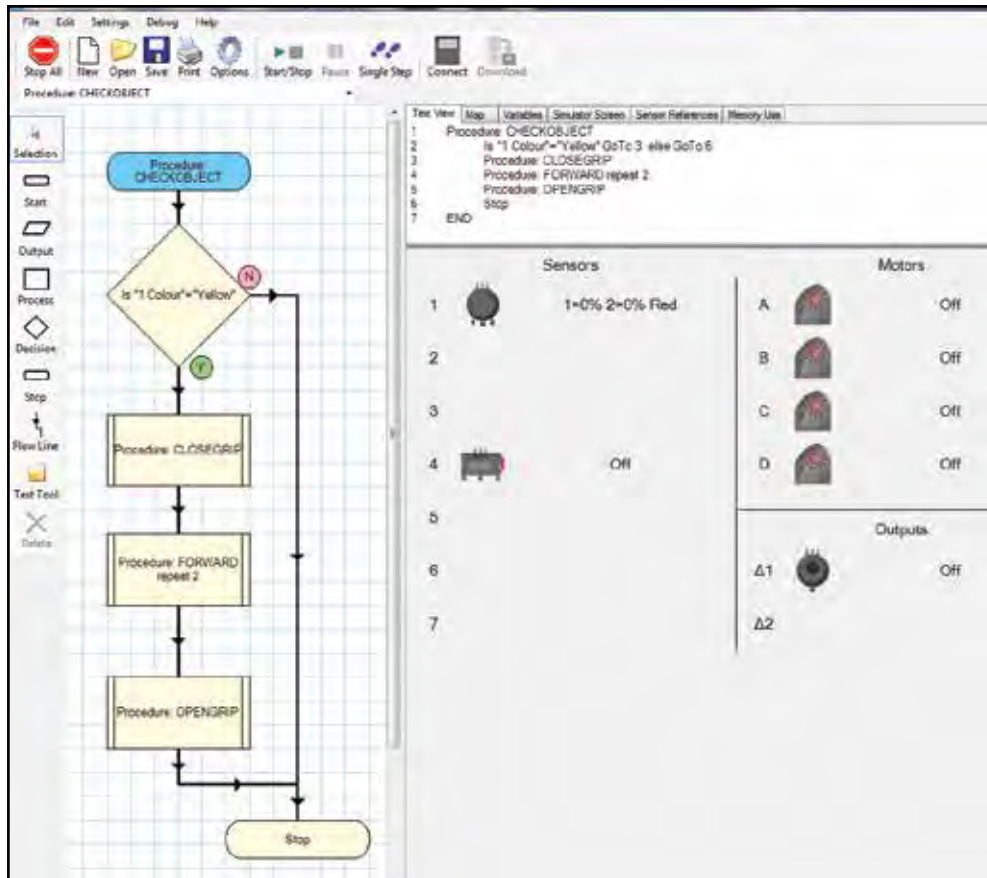


Figure 4a: Procedure "CHECKOBJECT".

Figure 1 (main program) shows a flow logic where the block with the procedure "CHECKOBJECT" is activated. Figure 4a shows the details of the flowchart logic for the "CHECKOBJECT" procedure.

- As seen in Figure 4a, the Colour/Light Sensor controls the function of the hand grip. (Later when you run the program, you will place a yellow ball in front of the Colour/Light Sensor.) "If" the sensor identifies the colour yellow, "then" it means the yellow K'NEX ball is ready to be grabbed and the procedure "CLOSEGRIP" is activated.
- When you place a "Decision" symbol in the program and click on it, you need to enter information in the input/output panel at the bottom right of the screen for the Colour/Light Sensor to respond to the colour yellow. From the "Sensor" pull down menu, select "1 Light 1," click "=" and click "yellow." The "Decision" symbol will send the program in one direction if the sensor identifies yellow and in another direction for any other colour.
- Figure 4b shows the logic of the "CLOSEGRIP" procedure that closes the Robot's hand to grab the ball. Motor A turns forward at 40% for 1 second to grip the ball.
- After the ball has been grabbed, the Figure 4a logic shows that the procedure "FORWARD" is activated and the Robot walks forward for a few steps. (This is the same "Forward" procedure you used earlier except the number of repeats has been changed to 2.) When the Robot stops walking the procedure "OPENGRIP" is activated and the ball is released.
- Figure 4c shows the "OPENGRIP" logic flow where the motor A is in reverse drive mode at 40% power for 1 second causing the robot to drop the ball.

Figure 1 (main program) has another block showing the procedure "HANDBUTTON". When this procedure is placed in the same loop as the "CHECKOBJECT" procedure, the program will check the condition of the Colour/Light Sensor to see if it is reading yellow and then check the Push Button to see if it is depressed. Figure 5 shows the logic for the procedure "HANDBUTTON". The Figure 5 logic strategy checks the condition of the pushbutton. If it is pressed, the procedure "OPENCLOSEGRIP" procedure (Figure 6) is activated. This procedure will open and close the hand three times.

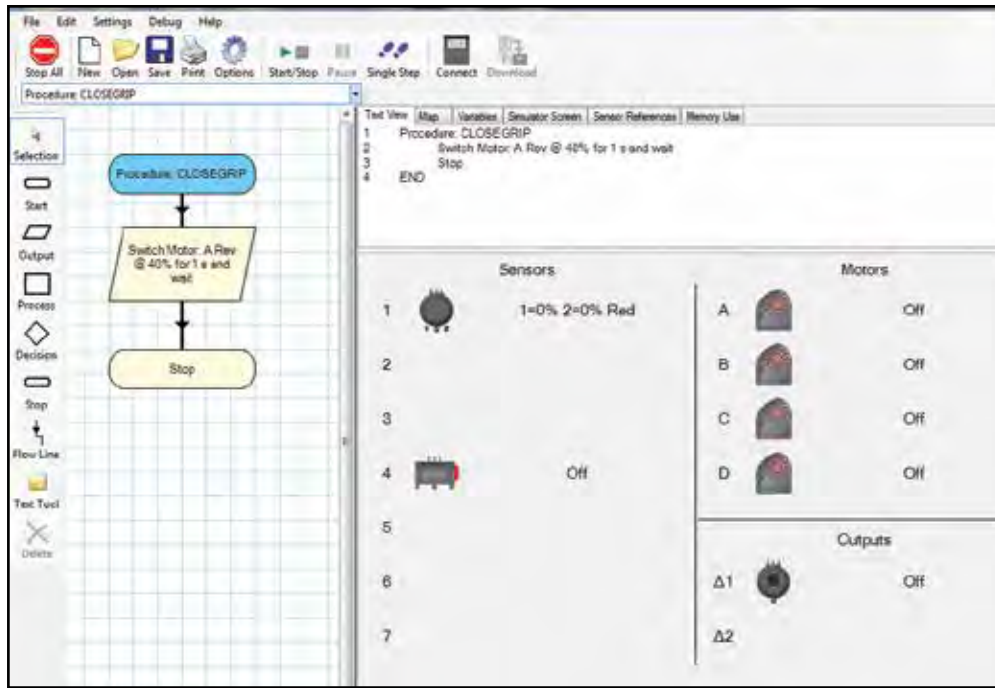


Figure 4b: Procedure "CLOSEGRIP".

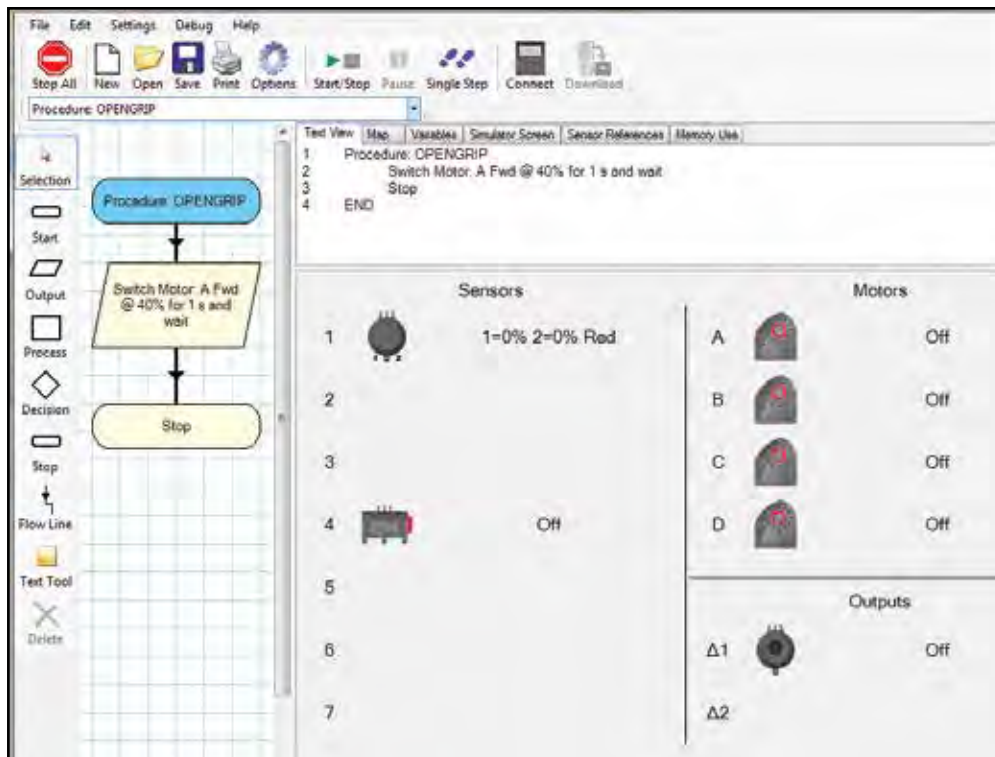


Figure 4c: Procedure "OPENGRIP".

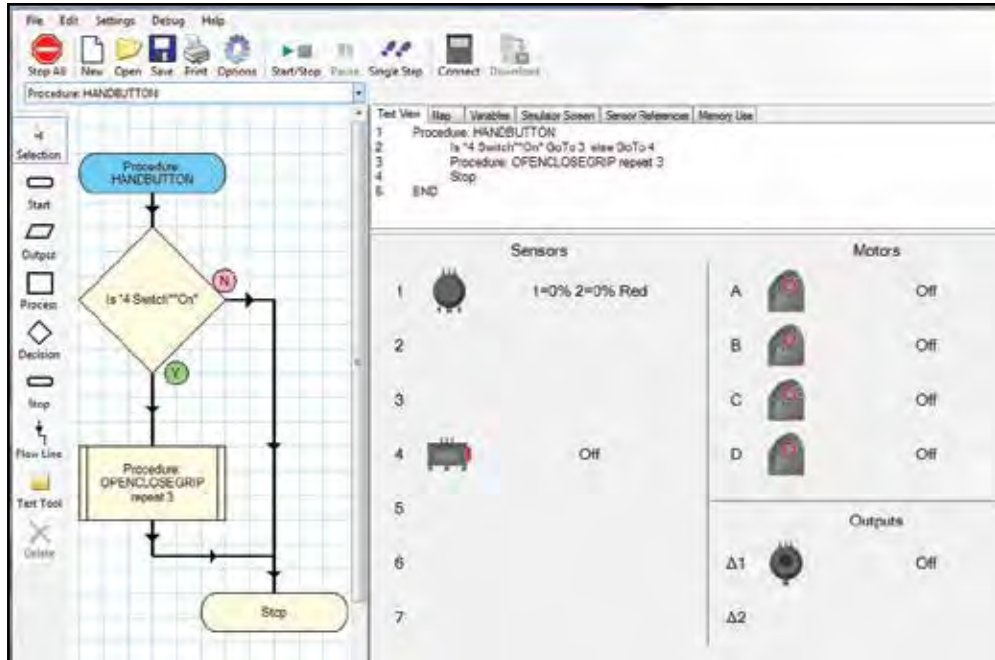


Figure 5: Procedure "HANDBUTTON".

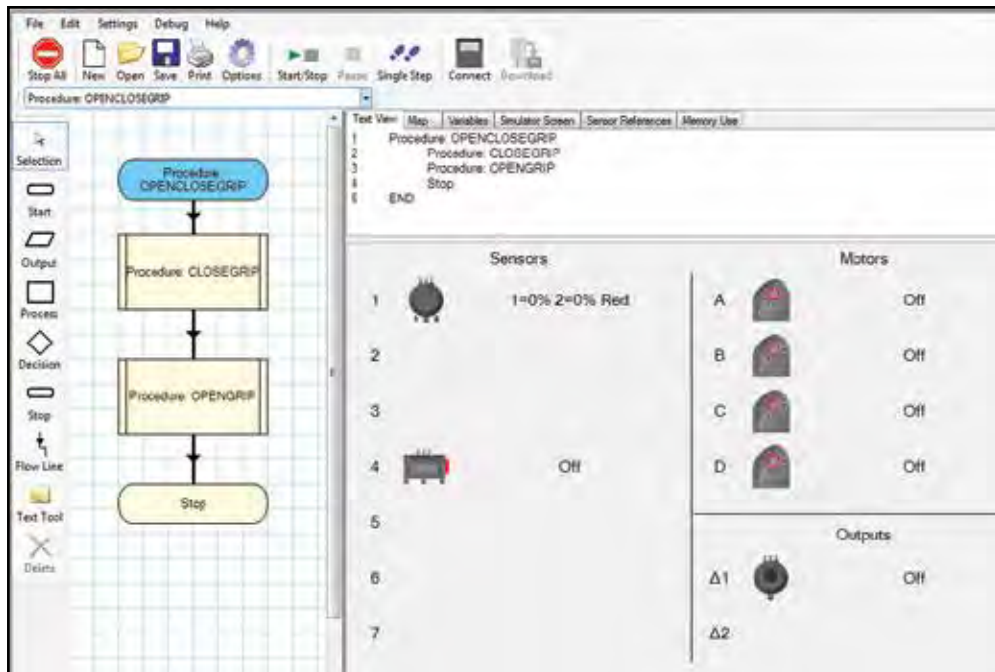


Figure 6: Procedure "OPENCLOSEGRIP".

If you have designed your Main Program and procedures carefully and identified appropriate values for your sensors, the Bi-Ped Robot should have operated as you expected. Make every effort to refine your program to optimise the actions of the Robot.

When you are satisfied with the operation of your model, move on to the challenges below or allow your imagination to guide you as you program your model to complete tasks of your own design.

Your **challenge** is to:

### **CHALLENGE 1**

Design and write a program and procedures using the Distance Sensor to stop the model when the Bi-Ped Robot is a specified distance from an object or a wall. For this activity, refer to the average digital value for the Distance Sensor from Student Response Sheet 1 before creating a procedure for STOP WHEN YOU SEE A WALL.

When you have designed your program, enter it on Response Sheet 1 or attach printed copies of the program and procedures to Response Sheet 1.

Test your model, troubleshoot its performance, and make changes to your procedures and program as necessary to improve the model's operation.

### **CHALLENGE 2**

Design and write a Main Program and procedures that utilises each of these sensors and inputs; Colour/Light Sensor, Push Button Sensor, and Distance Sensor to complete a task.

When you have designed your programs, enter them on Response Sheet 2 or attach printed copies of the program and procedures to Response Sheet 2.

Test your model, troubleshoot its performance, and make changes to your procedures and program as necessary to improve the model's operation.

### **CHALLENGE 3**

Redesign one of your main programs to incorporate the LED that is included in the head of the model.

When you have designed your programs, enter them on Response Sheet 2 or attach printed copies of the program and procedures to Response Sheet 2.

Test your model, troubleshoot its performance, and make changes to your procedures and program as necessary to improve the model's operation.



# Student Response Sheet 1

## STOP THERE'S A WALL!

Name \_\_\_\_\_ Group # \_\_\_\_\_

Describe the information that a flowchart design needs to contain in a procedure to check for an obstacle.

Write a flowchart design for the obstacle detection procedures.

**LESSON 3**  
**Student Response Sheet 1**  
**(continued)**

Name \_\_\_\_\_ Group # \_\_\_\_\_

Draw a complete main program flowchart design for the Bi-Ped Robot that incorporates your newly designed procedure(s).

## Student Response Sheet 2

# COLOUR/LIGHT SENSOR PUSH BUTTON & DISTANCE SENSOR

Name \_\_\_\_\_ Group # \_\_\_\_\_

**CHALLENGE** – Design and write a program and procedures that utilizes each of these sensors and inputs (Colour/Light Sensor, Push Button & Distance Sensor) to complete a task.

Describe the information that a flowchart design needs to contain in a procedure to complete the challenge above.

**LESSON 3**

**Student Response Sheet 2**  
**(continued)**

Name \_\_\_\_\_ Group # \_\_\_\_\_

Write a flowchart design for the procedure(s) that will meet the challenge.

Draw a complete main program flowchart design for the Bi-Ped Robot that incorporates your newly designed procedures.

### LESSON 3

## Student Response Sheet 3

# INCORPORATE AN LED

Name \_\_\_\_\_ Group # \_\_\_\_\_

Describe the information that a flowchart design needs to contain in a procedure to include the LED.

Write a flowchart design for the procedure that includes an LED.

Draw a complete main program flowchart design for the Bi-Ped Robot that incorporates your newly designed procedure(s).





## LESSON 4:

# Search & Rescue Robot



The Search & Rescue Robot is another interactive K'NEX robot. Students build and program its motors and sensors to interact with each other. In previous lessons the concept of motors, their performance and motion was introduced. In this lesson, the robot uses four motors: moves using two motors, moves its claws up and down using a third motor and grabs a ball and stand using a fourth motor. The robot has sensors that can send feedback signals to the controller which in turn will activate motors on the model. The Search & Rescue Robot is designed to travel to an object's location to pick up a ball on a stand, carry the ball and stand to a new location, and move away. This requires knowledge of different types of sensors to control the movement of the robot and the pick/grab action used to gather a ball. Students will begin by using the push button sensor to gather data that will direct the robot to close the claws to pick up and later release the ball and stand. The robot also includes a distance sensor that can be used to control its drive motors.

### Objectives

*Students will be able to:*

- Work in collaborative teams.
- Build the Search & Rescue Robot.
- Design and write a looped program, create procedures and call the procedures within the main program.
- Gain experience working with and programming a push button and distance sensor.
- Learn and practice strategies to calibrate motors and sensors.

### Materials

*Each group will need:*

- Built Search & Rescue Robot model  
(Go to [www.knex.com/robotics](http://www.knex.com/robotics) for instructions)
- PC and K'NEX Robotics Programming Software
- K'NEX Robotics User's Guide
- Student Instruction Sheet A – Lesson 4
- Copies of Student Response Sheets 1, 2, & 3 – Lesson 4

### Teacher's Notes

- Identify student groups of 2 - 3.
- Allow the students to explore the K'NEX Education materials and provide time to review the K'NEX Robotics User's Guide.
- Request that students show their progress before they begin to program their model.
- Students may need help as they program their models.

A partial, main program flowchart template is provided to help students organise their thought processes as they develop the procedures and symbols to complete the program.

## PROGRAMMING

The Search & Rescue Robot is designed to search for a ball on a stand, grab it in its claws and carry it to a new destination. At its destination, it will release the ball and stand and move away. These tasks require that the Robot perform the following functions:

- **Drive forward and reverse:** it will need two motors to drive forward, drive backward, and turn right or left.
- **Lift up and down:** it will need one motor to move the claws up and down.
- **Grab:** one motor can both open and close the claws.
- **Sensors:** the digital readings from a sensor sends input information to the controller that directs the motors to carry out the Robot's motions.

### Main Program:

Creating a main program was described in Lesson 1 and creating a procedure was described in Lesson 2. For more information, read the K'NEX Robotics User's Guide.

When the Robot was constructed, a Push Button Sensor was placed at the base of the claws. The push button will be pushed if it comes in contact with a ball and stand. The main program must include a loop that allows the Control Box to keep the model moving forward until the push button is pressed. The loop is initiated with a Decision symbol and a Flow Line that returns the program flow back to a point between the Open and Forward procedures. Once the Robot has begun to move forward, it will continue until the push button is pressed by, in this case, the ball and stand.

Once the push button is pressed, the procedure 'CLOSE' is activated and the claws close to grab the ball. After it grabs the object, the procedure 'UP' is initiated to lift it up the claws and 'LEFT' turns the robot to the left. 'FORWARD2' drives the robot forward for a specific distance. The robot carries the ball to its destination and drops it (procedure 'DOWN' and 'OPEN'). The Robot then drives away from the ball moving backwards (procedure 'BACKWARD'). Also, you will note the main program includes procedures to allow the LED to shine green when the ball is grabbed and shine red when the ball is released.

As this is the fourth lesson, students should be able to program the various procedures that are required to complete the main program.

## Student Instruction Sheet A – Lesson 4

As you create the following procedures, you will be introduced to the use of a push button as a sensor. This information and the procedures you design will allow you to program the Search & Rescue Robot to respond to sensor input. Sensor input sent to the controller will activate the various motors.

The task assigned to the Search & Rescue Robot is to approach a ball and stand, grab them, carry them to another location, release them and back away. When the ball and stand are in the claws of the Robot, a green LED is lit, when the ball and stand are released, the LED is lit.

### Requirements:

Complete Student Response Sheets 1, 2, & 3 as you work on this activity to keep a record of your work.

### Before you begin:

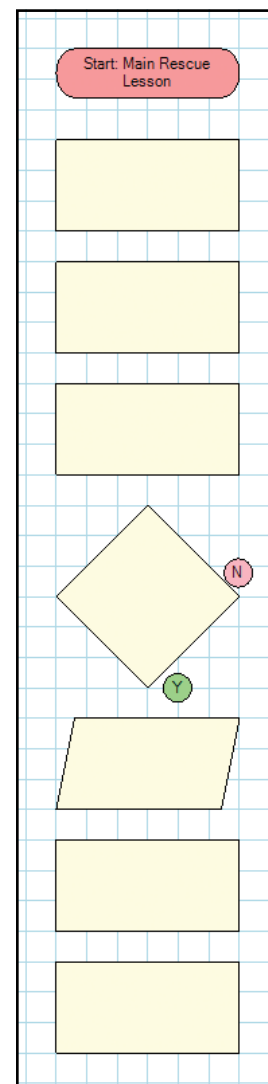
- Review the building instructions for the Search & Rescue Robot.
- Read and review the K'NEX Robotics User's Guide.  
Pay particular attention to creating a procedure within a program and how to include the procedure in the main program.

### Steps:

- Build the Search & Rescue Robot per the K'NEX Building Instructions.
- **DO NOT** forget to power the Control Box.
- Open the K'NEX Robotics Programming Software.
- Click the 'Connect' icon on the top menu bar to connect the computer and Control Box.
- **Ensure that the claws on your Robot are in the up and closed position.**
- **Place a ball on one of the stands you constructed and place it at some distance in front of the Search & Rescue Robot.**
- **Test the function of the motors, sensors, and outputs on your model.**  
**Enter the information requested on Student Response Sheet 1 and complete the rest of the sheet before you begin programming.**

Your task is to write a program to operate the Search & Rescue Robot. Use the template shown in Figure 1 to begin the main program. Name the main program 'Main – Rescue Lesson' and place blank Process, Decision, and Output symbols in place as shown. Additional symbols required to finish the program should be placed to the right side of the grid to keep the main program to a single screen. You will be prompted to add additional symbols when necessary.

The main program will include many procedure symbols that you must create in order to complete the main program. Some of these procedures will be easy and some will be more difficult. A few complete procedures will be shown in Figures 2, 3, and 4. Add flow lines to your main program as you complete the procedures and symbols that will make up the main program. The running times and revolutions of the motors in these procedures are based on sample programs prepared by K'NEX. Your particular running times and revolutions of the motors may vary from these printed values.



**Figure 1:** Partial Template for the 'Main – Rescue Lesson' Program.

Program the following procedures to complete the flowchart that will enable your model to operate. Refer to Student Response Sheet 2 as you program to answer questions related to specific procedures and programming strategies.

**Procedure Down:**

(This procedure will appear in the first Process symbol on the Figure 1 template. Succeeding procedures will follow in order.)

This procedure moves the Robot arms/claws down. Name the procedure and program it to slowly move the claws to their down position. You will need to determine which motor raises and lowers the claw, the direction to turn the motor, a power percentage for the motor, and the number of revolutions the motor will turn. (HINT: the motor will turn less than one revolution and you will need to have the procedure 'Wait' until the motor stops.)

**Procedure Open:**

This procedure opens the Robot claws. (HINT: Remember that the motor percentage and the number of revolutions may vary slightly from those in the Output symbol.) See Figure 2.

**Procedure Forward:**

This procedure will power both motors in the same direction to move the Robot slowly forward. Do not include any of the following in the Process symbol inside this procedure: "s" or "Rev" or "Wait."

**Decision Symbol:**

Program the symbol to determine if the push button is on. The tricky part is placing the Flow Lines for this symbol.

**Output Symbol:**

Program the symbol to turn the LED on green. (HINT: Check to see where the LED is plugged in.)

**Procedure Close:**

Program the procedure to close the claws. (HINT: Can you reverse the information you programmed in the 'Open' procedure?)

**Procedure Up:**

This procedure raises the Robot claws. See Figure 3.

**Procedure Left:**

This procedure turns the Robot to the left. Notice the motors only turn only 0.54 revolutions to complete the turn and the motors move at slightly different speed settings. See Figure 4.

**Procedure Forward 2:**

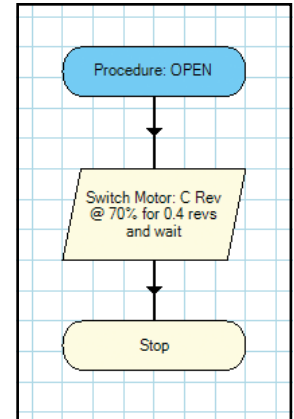
This procedure moves the Robot forward for just 1.3 revolutions and waits for the motors to complete the revolutions. Check your previous 'Forward' procedure to determine the power settings for the motors to move in a straight line.

**Procedure Down:**

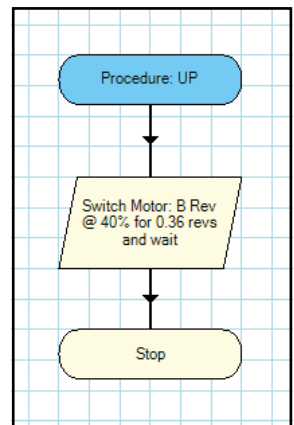
You programmed this procedure earlier. Place your Process Symbol and call the 'Down' Procedure again.

**Procedure Open:**

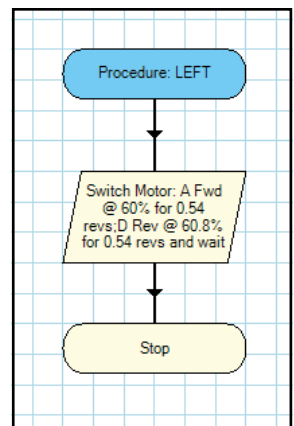
You programmed this procedure earlier. Just place your Process symbol and call the 'Open' procedure again.



**Figure 2:**  
Procedure Open.



**Figure 3:**  
Procedure Up.



**Figure 4:**  
Procedure Left.

**Output Symbol:**

Program the symbol to turn the red LED on.

**Procedure Backward:**

This procedure moves the Robot backwards for two revolutions of the motors and waits until the motors have completed their task.

**Output Symbol:**

Turn the red LED off.

If you have all of these procedures in place, you should be able to complete the main program and test your Robot.

**CHALLENGE ACTIVITY:**

Your challenge is to design and write a main program and procedures using a Distance Sensor for the Search & Rescue Robot. The Robot must pick up a ball with its stand and drop it in a new location. *(You may also use the push button and LED in your main program.)*

When you have designed your program and procedures, enter them on Response Sheet 3.

Test your model, troubleshoot its performance, and make changes to your procedures and main program as necessary to improve the Robot's operation and performance.

**SUPER CHALLENGE ACTIVITY:**

Your challenge is to program the Search & Rescue Robot to pick up and move two or even three balls and stands with a single main program. Good Luck!



# Student Response Sheet 1

Name \_\_\_\_\_

Group # \_\_\_\_\_

List the location of the various motors, sensors, and outputs.  
Also list what each does to support your program.

<b>MOTORS:</b>	<b>ACTION: What does each item do?</b>
Motor A	
Motor B	
Motor C	
Motor D	
<b>SENSORS:</b>	
Push Button	
Distance	
<b>OUTPUT:</b>	
LED	

Explain how the information on the chart above will be of assistance as you program your model.

# Student Response Sheet 2

Name \_\_\_\_\_

Group # \_\_\_\_\_

**Procedure Down:**

Why is it important to move the claws down slowly?

**Procedure Forward:**

The Process symbol for this procedure only includes two motor designations and power percentages for each. There is no motor running time, number of revolutions, or wait for the motor(s) to finish commands included in the Process symbol. Why is this strategy necessary for the operation of this procedure?

**Decision Symbol:**

Draw a diagram to show where you placed the Flow Lines for this symbol. When you tested the program did the Robot respond to the Push Button as you expected? If not, what changes did you have to make to the Decision symbol or Flow Lines to correct the situation?

# Student Response Sheet 3

Name \_\_\_\_\_ Group # \_\_\_\_\_

Design and write a main program and procedures that utilise the Distance Sensor on the Search & Rescue Robot. The Robot must pick up a ball with its stand and drop it in a new location. *(You may also use the Push Button and LED in your main program.)*

Describe how your main program operates in paragraph form. Include information about how you utilise the Distance Sensor to assist the Robot in completing its task. Also, describe any problems you encountered during the programming and/or testing of the Robot. How did you correct these problems?

Print a copy of your main program and procedures for this challenge and attach them to this Student Response Sheet.



## LESSON 5: Ball Factory



Robotic systems are of many types; some are small, some are drivable, but some are also huge as we often see in industrial settings. A simple example is the Ball Factory. The Ball Factory will give students a perspective of how factory conveyor systems function as part of a quality control process in an industrial setting. Students will gain experience as they build a system to lift, pick up a ball, sense the colour of the ball, decide if the ball needs to be eliminated from the system and much more. Previous lessons performed communication tasks between the motors and sensors. Here students will continue to do the same with more checks as the balls move through four separate stations that make up the Ball Factory.

The motion of balls from one station to the other with quality control checks along the way requires an interactive program where processes depend on previous actions of the system. Programming the Ball Factory system will prepare students to take on challenging logic development activities and communicate with multiple sensors and motors in a complex program. The feedback sensor system feature of this lesson is more robust than previous lessons. It continuously checks sensor readings to determine if the ball travelling on the conveyor is the correct colour (yellow: in this lesson) and if it is not, the ball is removed from the system. In simple terms the sensors act like eyes and send check signals back to the Control Box, which in turn sends commands to the motors to operate the Ball Factory.

### Objectives

*Students will be able to:*

- Work in collaborative teams.
- Build the Ball Factory following the Building Instructions.
- Explore the various mechanical functions of the Ball Factory such as lift, pick and drop, rotational and translational motion.
- Design and write a loop program, create procedures and call the procedures into the main (flowchart) program.
- Gain advanced experience using a feedback sensor system.

### Materials

*Each group will need:*

- Built Ball Factory model (*Go to [www.knex.com/robotics](http://www.knex.com/robotics) for instructions*)
- A red, green and yellow K'NEX ball
- Table/counter space to operate and store the model
- PC Computer and K'NEX Robotics Programming Software
- K'NEX Robotics User's Guide
- Student Instruction Sheet A – Lesson 5
- Copies of Student Response Sheets 1, 2, & 3 – Lesson 5

## Teacher's Notes

- Identify student groups of 2 - 3.
- Allow the students to explore the K'NEX Education materials and give them time to read the sections of the K'NEX Robotics Programming Software User's Guide related to procedures and incorporating those procedures within a main program.
- Provide time for students to build the Ball Factory.
- Request that students show their progress before they begin to program their model.
- Ensure students test their motors to determine the directions they will move when programmed.
- Instruct students to begin their programming.

By this point students should have developed programming skills that will allow them to produce their own Main Program and Procedures for the Ball Factory.

## Ball Factory Concept

The feedback system the students program to communicate between the Control Box and sensors will allow balls to travel smoothly and be sorted as necessary based on their colour.

### *Station #1:*

A ball is placed at the bottom of the chain drive escalator. The Distance Sensor identifies that a ball is present and the Control Box starts the motor on the chain drive to lift the ball to a ramp that carries the ball down to the next station.

### *Station #2:*

The ball drops into position near a Colour/Light Sensor. The sensor determines the colour of the ball and transmits that information to the Control Box.

OPTION 1: If the ball is yellow, a motor turns a wheel that pushes the ball forward along another ramp to an area where it can be picked up.

OPTION 2: If the ball is not yellow, a trap door opens to cause the ball to drop out of the Ball Factory as it is pushed forward.

### *Station #3*

A mechanical arm swings a claw over the top of the yellow ball, drops to capture the ball and carries the ball to a delivery area that allows the ball to travel through the Ball Factory again.

Four motors have been included to power the different systems on the Ball Factory. The Control Box will receive data from three different sensors and use that information to direct the motors. Students will also add an LED to the Ball Factory. When all of these devices are programmed, the Ball Factory will operate to move a yellow ball continuously through the system. Balls that are not yellow will be rejected and dropped out of the Ball Factory.



## Student Instruction Sheet A – Lesson 5

The following activities will introduce you to the use of multiple sensors in robotic systems and will allow you to program the Ball Factory. The program you design will move different colour K'NEX balls one at a time through the Ball Factory using four motors. The motors will be directed by the Control Box based on sensor data that is collected as the ball moves from one location to another. You will be successful when a yellow ball moves continuously through the Ball Factory and red or green balls are rejected and removed from the Factory.

### Requirements:

Complete Student Response Sheets 1 & 2 as you work on this activity to keep a record of your work.

### Before you begin:

- Review the building instructions for the Ball Factory before building.
- Read and review the K'NEX Robotics User's Guide. Pay particular attention to creating a "procedure" within a program and how to include the procedure in the main program.
- Review the operation and programming of the Colour/Light Sensor, Distance Sensor and Push Button Sensor.

### Steps:

1. Build the Ball Factory as shown in the Building Instructions.
2. Collect a red, green and yellow K'NEX ball.
3. Power the Control Box.
4. Determine the action of each of the motors, sensors and outputs and record that information on Student Response Sheet 1.
5. Complete Student Response Sheet 2 as you complete this lesson.
6. Move on to the programming section below for further instructions.

## PROGRAMMING

### **ACTION 1:**

To operate the Ball Factory, place balls one at a time at the bottom of the chain drive escalator. There is a Distance Sensor at this point that will be used to start the balls moving. Your first task is to prepare a procedure that will identify that there is a ball at the base of the chain drive. If so, activate a motor to power the chain drive long enough to raise the ball to the ramp that runs down one side of the Ball Factory. Check the operation of your procedure several times to troubleshoot its operation. Also, ensure that the procedure places the ball so it will roll down the ramp. Add an LED that indicates the Ball Factory is operating.

### **ACTION 2:**

When the ball reaches the end of the ramp, it will drop into a "V" shaped space formed by pairs of white rods on a wheel. The Colour/Light Sensor shines its light on the ball as it determines its colour. Add a check to the program to determine if the ball is yellow. *(NOTE: a Decision Symbol will allow you to continue now and return to the symbol later to program what will happen if the ball is not yellow.)*

## LESSON 5 – Student Instruction Sheet A (continued)

### **ACTION 3:**

Write a procedure that will advance the ball in the system if the ball is yellow. The procedure will operate a motor driven wheel to push the ball down another ramp which leads to a catch area that will stop the ball. Continue by programming the motor that controls the mechanical arm to swing over the ball, drop down and grasp the ball with its claw. Notice that the mechanical arm presses a push button as it comes down on the ball.

### **ACTION 4:**

Program a check (Decision symbol) to determine if the push button is pressed. If so, direct the motor controlling the mechanical arm to rotate the arm through a 180 degree flip. This will return the yellow ball to the start of the loop, where it will begin moving through the Ball Factory again.

### **ACTION 5:**

Return to the Decision Block that checked to see if the ball in the system was yellow. Write a procedure that will reject red and green balls that may pass through the Ball Factory. If the ball is red or green, program the motor which operates the trapdoor in the ramp to open the door. The door should be opened before the wheel that advances the ball out of the Colour/Light Sensor is activated. The LED colour should be changed during this process to indicate a ball has been rejected from the Ball Factory. The Main program should restart at this time.

### **ACTION 6:**

Ensure that your Flow Lines are organized so the program is able to run continuously.

# Student Response Sheet 1

Name \_\_\_\_\_

Group # \_\_\_\_\_

List the location of the various motors, sensors, and outputs.  
Also list what each does to support your program.

<b>MOTORS:</b>	<b>ACTION: What does each item do?</b>
Motor A	
Motor B	
Motor C	
Motor D	
<b>SENSORS:</b>	
Push Button	
Distance	
<b>OUTPUT:</b>	
LED	

Explain how the information on the chart above will be of assistance as you program your model.

# Student Response Sheet 2

Name \_\_\_\_\_ Group # \_\_\_\_\_

You programmed 6 different actions (*and possibly more*) to complete this lesson. What problems did you encounter as you programmed each of these actions? Indicate how you were able to overcome these issues.

**ACTION 1:**

**ACTION 2:**

**LESSON 5**  
**Student Response Sheet 2**  
**(continued)**

Name \_\_\_\_\_

Group # \_\_\_\_\_

***ACTION 3:***

***ACTION 4:***

**LESSON 5**  
**Student Response Sheet 2**  
**(continued)**

Name \_\_\_\_\_

Group # \_\_\_\_\_

***ACTION 5:***

***ACTION 6:***