Lesson 3.1 Machine Control (VEX) - Overview

Preface

From iPods to automobiles, we use computers every day. Computers are sometimes so small and hidden that we don't even realize we're using a computer. Many of us never think about automobiles containing computers; however, today's vehicles are packed with tiny computers that regulate and monitor systems such as air bags and cruise control. How much more control will computers take from drivers in the future? What will drivers be willing to let their cars do for them? With GPS systems that provide routes and track speed, what are the barriers for autonomous cars?

In this lesson students will learn how to control mechanical processes using computer software and hardware. The software communicates through a hardware interface with different inputs and outputs.

Understandings

1. Flowcharts provide a step by step schematic representation of an algorithm or process.
2. Control systems are designed to provide consistent process control and reliability.
3. Control system protocols are an established set of commands or functions typically created in a computer programming language.
4. Closed loop systems use digital and analog sensor feedback to make operational and process decisions.
5. Open loop systems use programming constants such as time to make operational and process decisions.

Knowledge and Skills

It is expected that students will:

- Create detailed flow charts that utilize a computer software application.
- Create control system operating programs that utilize computer software.
- Create system control programs that utilize flowchart logic.
- Choose appropriate input and output devices based on the need of a technological system.
- Differentiate between the characteristics of digital and analog devices.
- Judge between open and closed loop systems in order to choose the most appropriate system for a given technological problem.
- Design and create a control system based on given needs and constraints.

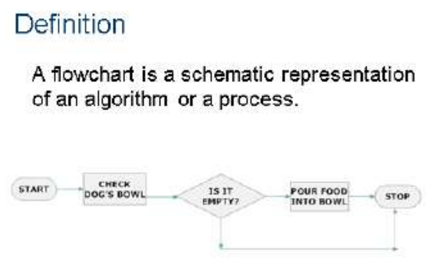
Essential Questions

1. What are the advantages and disadvantages of using programmable logic to control machines versus monitoring and adjusting processes manually?
2. What are some everyday seemingly simple devices that contain microprocessors, and what function do the devices serve?
3. What questions must designers ask when solving problems in order to decide between digital or analog systems and between open or closed loop systems?
4. # Lesson 3.1 Machine Control - Key Terms

Term	Definition
Algorithm	A step-by-step procedure for solving a problem or accomplishing some end, especially by

	computer.
Analog Signal	A signal having the characteristic of being continuous and changing smoothly over a giver rather than switching suddenly between certain levels.
Analog to Digital	Conversion of an analog signal to a digital quantity such as binary.
Closed Loop System	A control system that considers the output of a system and makes adjustments based on output.
Data	Information encoded in a digital form, which is usually stored in an assigned address of a memory for later use by the processor.
Digital Signal	A system of discrete states: high or low, on or off, 1 or 0.
Digital to Analog	Conversion of a digital signal to its analog equivalent, such as a voltage.
Electromagnet	A conductor wrapped around an iron core. The two ends of the conductor are attached to source. When current passes through the conductor, the iron core becomes magnetized.
Feedback	The return to the input of a part of the output of a machine, system, or process (as for proc changes in an electronic circuit that improve performance or in an automatic control devic provide self-corrective action).
Flowchart	A diagram that shows step-by-step progression through a procedure or system especially connecting lines and a set of conventional symbols.
Input	Information fed into a data processing system or computer.
Interface	The place at which independent and often unrelated systems meet and act on or commun with each other.
Microprocessor	The central processing unit that is generally made from a single integrated circuit.
Normally Closed	The contact of a relay that is closed when the coil is de-energized.
Normally Open	The contact of a relay that is open when the coil is de-energized.
NTC Resistor	A negative temperature coefficient, also known as a thermistor, is a sensitive resistor who primary function is to exhibit a change in electric resistance with a change in temperature.
Open Loop System	A control circuit in which the system output has no effect on the control.
Output	The information produced by a computer.
Photocell	A photo-sensitive resistor whose resistance decreases as the light striking the unit increas
Polarity	The type of charge an atomic particle has.
Potentiometer	A switch that can provide variable motion control. It can vary the resistance within the swit which affects both the current and voltage flowing out of the switch.
Programmable Logic Controller	A specialized heavy-duty computer system used for process control in factories, chemical and warehouses. Closely associated with traditional relay logic. Also called a programmab controller (PC).
Reed Switch	An electromagnetically operated switching device.

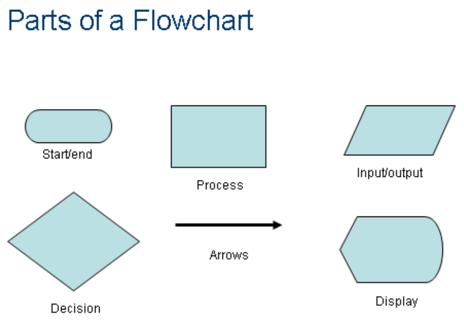
Sensor	A device that responds to a physical stimulus (as heat, light, sound, pressure, magnetism particular motion) and transmits a resulting impulse (as for measurement or operating a c
Subroutine	A subordinate routine; specifically, a sequence of computer instructions for performing a s task that can be used repeatedly.
Switch	A device for making, breaking, or changing the connections in an electrical circuit.
Transistor	A solid-state switching device.



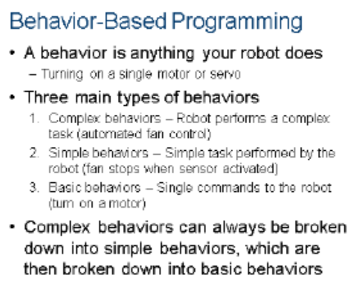
Definition

A flowchart is a schematic representation of an algorithm or a process.

START
CHECK DOG'S BOWL
IS IT EMPTY?
POUR FOOD INTO BOWL
STOP

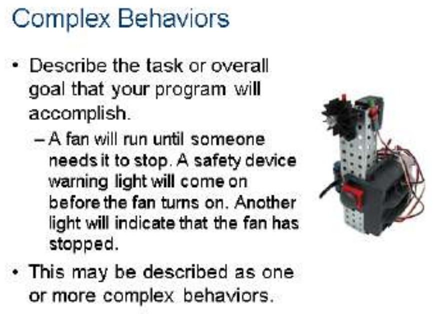


Parts of a Flowchart



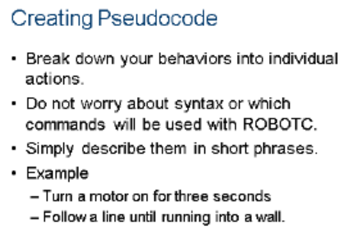
Behavior-Based Programming

- A behavior is anything your robot does
  - Turning on a single motor or servo
- Three main types of behaviors
  1. Complex behaviors – Robot performs a complex task (automated fan control)
  2. Simple behaviors – Simple task performed by the robot (fan stops when sensor activated)
  3. Basic behaviors – Single commands to the robot (turn on a motor)
- Complex behaviors can always be broken down into simple behaviors, which are then broken down into basic behaviors



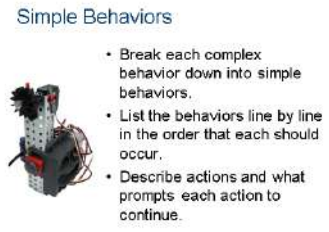
Complex Behaviors

- Describe the task or overall goal that your program will accomplish.
  - A fan will run until someone needs it to stop. A safety device warning light will come on before the fan turns on. Another light will indicate that the fan has stopped.
- This may be described as one or more complex behaviors.



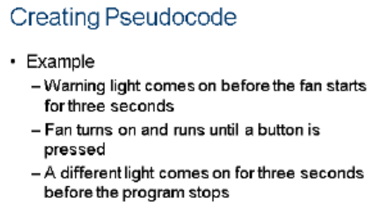
Creating Pseudocode

- Break down your behaviors into individual actions.
- Do not worry about syntax or which commands will be used with ROBOTC.
- Simply describe them in short phrases.
- Example
  - Turn a motor on for three seconds
  - Follow a line until running into a wall.



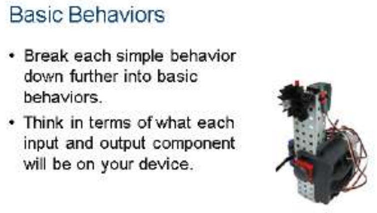
Simple Behaviors

- Break each complex behavior down into simple behaviors.
- List the behaviors line by line in the order that each should occur.
- Describe actions and what prompts each action to continue.



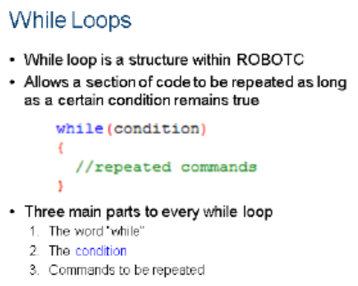
Creating Pseudocode

- Example
  - Warning light comes on before the fan starts for three seconds
  - Fan turns on and runs until a button is pressed
  - A different light comes on for three seconds before the program stops



Basic Behaviors

- Break each simple behavior down further into basic behaviors.
- Think in terms of what each input and output component will be on your device.

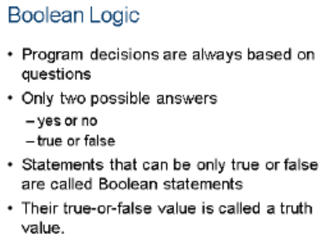


While Loops

- While loop is a structure within ROBOTC
- Allows a section of code to be repeated as long as a certain condition remains true

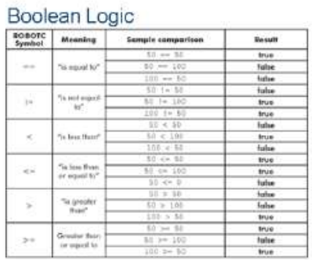
```
while(condition)
{
  //repeated commands
}
```

- Three main parts to every while loop
  1. The word "while"
  2. The condition
  3. Commands to be repeated



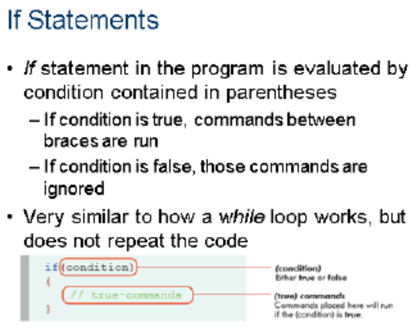
Boolean Logic

- Program decisions are always based on questions
- Only two possible answers
  - yes or no
  - true or false
- Statements that can be only true or false are called Boolean statements
- Their true-or-false value is called a truth value.



Boolean Logic

ROBOTC Symbol	Meaning	Sample comparison	Result
==	"is equal to"	50 == 50	true
		50 == 100	false
		100 == 50	false
!=	"is not equal to"	50 != 50	false
		50 != 100	true
		100 != 50	true
<	"is less than"	50 < 50	false
		50 < 100	true
		100 < 50	false
<=	"is less than or equal to"	50 <= 50	true
		50 <= 100	true
		50 <= 0	false
>	"is greater than"	50 > 50	false
		50 > 100	false
		100 > 50	true
>=	Greater than or equal to	50 >= 50	true
		50 >= 100	false
		100 >= 50	true

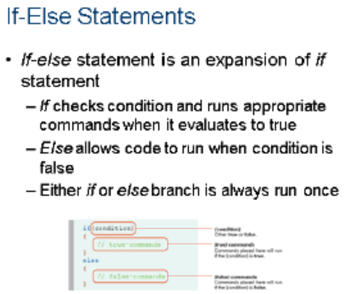


If Statements

- *If* statement in the program is evaluated by condition contained in parentheses
  - If condition is true, commands between braces are run
  - If condition is false, those commands are ignored
- Very similar to how a *while* loop works, but does not repeat the code

```
if(condition)
{
    // true-commands
}
```

(condition)
Either true or false
(true) commands
Commands placed here will run if the (condition) is true.



If-Else Statements

- *If-else* statement is an expansion of *if* statement
  - *If* checks condition and runs appropriate commands when it evaluates to true
  - *Else* allows code to run when condition is false
  - Either *if* or *else* branch is always run once

```
if(condition)
{
    // true-commands
}
else
{
    // false-commands
}
```

(condition)
Either true or false.
(true) commands
Commands placed here will run if the (condition) is true.
(false) commands
Commands placed here will run if the (condition) is false.

Many devices function without ever knowing whether they are doing the job that they were programmed to do. They might run for a specific amount of time or perform one function and then stop. For example if you set the clothes dryer to run for 45 minutes, your clothes might be dry or they might not be dry. A clothes dryer is an open loop system because the process provides no feedback to the device. Newer clothes dryers possess moisture sensors. The moisture sensors inform the machine when the clothes are dry, at which point the dryer can stop running. The feedback provided by the sensor makes this a closed loop system