

ROBOTC Natural Language - VEX Cortex Quick Reference:

<p>Set Servo Set a servo to a desired position. <i>Default servo and position: port6, 0.</i></p>	<pre>setServo();</pre>	<pre>setServo(port7, 95);</pre>
<p>Start Motor Set a specific motor to a speed. <i>Default motor and speed: port6, 95.</i></p>	<pre>startMotor(); wait(); stopMotor();</pre>	<pre>startMotor(port8, -32); wait(0.5); stopMotor(port8);</pre>
<p>Stop Motor Stop a specific motor. <i>Default motor: port6.</i></p>	<pre>startMotor(); wait(); stopMotor();</pre>	<pre>startMotor(port8, -32); wait(0.5); stopMotor(port8);</pre>
<p>Wait Wait an amount of time measured in seconds. <i>Default time: 1.0.</i></p>	<pre>startMotor(); wait(); stopMotor();</pre>	<pre>startMotor(port8, 63); wait(2.7); stopMotor(port8);</pre>
<p>Wait in Milliseconds Wait an amount of time measured in milliseconds. <i>Default time: 1000.</i></p>	<pre>startMotor(); waitInMilliseconds(); stopMotor();</pre>	<pre>startMotor(port8, 63); waitInMilliseconds(2700); stopMotor(port8);</pre>
<p>Until Touch The robot waits for the Touch Sensor to be pressed. <i>Default sensor port: dgt6.</i></p>	<pre>startMotor(); untilTouch(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilTouch(dgt110); stopMotor(port8);</pre>
<p>Until Release The robot waits for the Touch Sensor to be released. <i>Default sensor port: dgt6.</i></p>	<pre>startMotor(); untilRelease(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilRelease(dgt110); stopMotor(port8);</pre>
<p>Until Bump The robot waits for the Touch Sensor to be pressed in and then released out. <i>Default sensor port and delay time: dgt6, 10.</i></p>	<pre>startMotor(); untilBump(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilBump(dgt110, 100); stopMotor(port8);</pre>
<p>Until Button Press The robot waits for a button on the VEX LCD to be pressed. <i>Default button: centerBtnVEX.</i></p>	<pre>startMotor(); untilButtonPress(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilButtonPress(rightBtnVEX); stopMotor(port8);</pre>
<p>Until Sonar - Less Than The robot waits for the Sonar Sensor to read a value in cm less than the threshold. <i>Default threshold and sensor port: 30, dgt18+9.</i></p>	<pre>startMotor(); untilSonarLessThan(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilSonarLessThan(45, dgt12); stopMotor(port8);</pre>
<p>Until Sonar - Greater Than The robot waits for the Sonar Sensor to read a value in cm greater than the threshold. <i>Default threshold and sensor port: 30, dgt18+9.</i></p>	<pre>startMotor(); untilSonarGreaterThan(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilSonarGreaterThan(45, dgt12); stopMotor(port8);</pre>

ROBOTC Natural Language - VEX Cortex Quick Reference:

<p>Until Potentiometer - Greater Than The robot waits for the Potentiometer Sensor to read a value greater than a set position. <i>Default threshold and sensor port: 2048, in6.</i></p>	<pre>startMotor(); untilPotentiometerGreaterThan(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilSonarGreaterThan(4000, in4); stopMotor(port8);</pre>
<p>Until Potentiometer - Less Than The robot waits for the Potentiometer Sensor to read a value less than a set position. <i>Default threshold and sensor port: 2048, in6.</i></p>	<pre>startMotor(); untilPotentiometerLessThan(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilSonarLessThan(40, in4); stopMotor(port8);</pre>
<p>Until Dark The robot waits for the Light Sensor to read a value less than the threshold. <i>Default threshold and sensor port: 505, in2.</i></p>	<pre>startMotor(); untilDark(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilDark(1005, in4); stopMotor(port8);</pre>
<p>Until Light The robot waits for the Light Sensor to read a value greater than the threshold. <i>Default threshold and sensor port: 505, in2.</i></p>	<pre>startMotor(); untilLight(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilLight(1005, in4); stopMotor(port8);</pre>
<p>Until Rotations The robot waits for an encoder to reach a specified number of rotations. <i>Default rotations, encoder: 1.0, dgt11+2</i></p>	<pre>startMotor(); untilRotations(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilRotations(2.75, dgt13); stopMotor(port8);</pre>
<p>Until Encoder Counts The robot waits for an encoder to reach a specified number of encoder counts. <i>Default counts, encoder: 360, dgt11+2.</i></p>	<pre>startMotor(); untilEncoderCounts(); stopMotor();</pre>	<pre>startMotor(port8, 63); untilEncoderCounts(990, dgt13); stopMotor(port8);</pre>
<p>LED ON Turn an LED in a specified digital port ON. <i>Default sensor port: dgt12.</i></p>	<pre>turnLEDOn(); wait(); turnLEDOff();</pre>	<pre>turnLEDOn(dgt17); wait(0.5); turnLEDOff(dgt17);</pre>
<p>LED OFF Turn an LED in a specified digital port OFF. <i>Default sensor port: dgt12.</i></p>	<pre>turnLEDOn(); wait(); turnLEDOff();</pre>	<pre>turnLEDOn(dgt17); wait(0.5); turnLEDOff(dgt17);</pre>
<p>Flashlight ON Turn a VEX Flashlight in a specified motor port ON at a specified brightness. <i>Default motor port and brightness: port4, 63.</i></p>	<pre>turnFlashlightOn(); wait(); turnFlashlightOff();</pre>	<pre>turnFlashlightOn(port10, 127); wait(0.5); turnFlashlightOff(port10);</pre>
<p>Flashlight OFF Turn a VEX Flashlight in a specified motor port OFF. <i>Default motor port: port4.</i></p>	<pre>turnFlashlightOn(); wait(); turnFlashlightOff();</pre>	<pre>turnFlashlightOn(port10, 127); wait(0.5); turnFlashlightOff(port10);</pre>
<p>Robot Type Choose which robot you are using (Recbot or Swervebot). <i>Default bot: none.</i></p>	<pre>robotType();</pre>	<pre>robotType(swervebot);</pre>

ROBOTC Natural Language - VEX Cortex Quick Reference:

Forward The robot drives straight forward. <i>Default speed: 95.</i>	<pre>forward(); wait(); stop();</pre>	<pre>forward(63); wait(2.0); stop();</pre>
Backward The robot drives straight backward. <i>Default speed: -95.</i>	<pre>backward(); wait(); stop();</pre>	<pre>backward(63); wait(2.0); stop();</pre>
Point Turn The robot makes a sharp turn in place. <i>Default direction and speed: right, 95.</i>	<pre>pointTurn(); wait(); stop();</pre>	<pre>pointTurn(left, 63); wait(0.4); stop();</pre>
Swing Turn The robot makes a wide turn, activating only one drive motor. <i>Default direction and speed: right, 95.</i>	<pre>swingTurn(); wait(); stop();</pre>	<pre>swingTurn(left, 63); wait(0.75); stop();</pre>
Stop The robot halts both driving motors, coming to a stop.	<pre>forward(); wait(); stop();</pre>	<pre>forward(63); wait(2.0); stop();</pre>
Line Track - for Time The robot tracks a dark line on a light surface for a specified time in seconds. <i>Default time, threshold, sensors: 5.0, 505, in1, in2, in3 (Left, Center, Right).</i>	<pre>lineTrackForTime(); stop();</pre>	<pre>lineTrackForTime(7.5, 99, in6, in7, in8); stop();</pre>
Line Track - for Rotations The robot tracks a dark line on a light surface for a specified distance in rotations. <i>Default time, threshold, sensors: 3.0, 505, in1, in2, in3 (Left, Center, Right).</i>	<pre>lineTrackForRotations(); stop();</pre>	<pre>lineTrackForRotations(4.75, 99, in6, in7, in8); stop();</pre>
Move Straight - for Time The robot will use encoders to maintain a straight path for a specified time in seconds. <i>Default time, rightEncoder, leftEncoder: 5.0, dgt1+2, dgt3+4.</i>	<pre>moveStraightForTime(); stop();</pre>	<pre>moveStraightForTime(7.5, dgt15, dgt13); stop();</pre>
Move Straight - for Rotations The robot will use encoders to maintain a straight path for a specified distance in encoder rotations. <i>Default rotations, rightEncoder, leftEncoder: 1.0, dgt1+2, dgt3+4.</i>	<pre>moveStraightForRotations(); stop();</pre>	<pre>moveStraightForRotations(4.75, dgt15, dgt13); stop();</pre>
Tank Control The robot is remote controlled with the right motor mapped to the right joystick and the left motor mapped to the left joystick. <i>Default right and left joystick: Ch2, Ch3.</i>	<pre>while(true) { tankControl(); }</pre>	<pre>while(true) { tankControl(Ch1, Ch4); }</pre>
Arcade Control The robot is remote controlled with both motors mapped to a single joystick. <i>Default vertical and horizontal joysticks: Ch2, Ch1.</i>	<pre>while(true) { arcadeControl(); }</pre>	<pre>while(true) { arcadeControl(Ch1, Ch4); }</pre>