

# ROBOTC Natural Language - VEX PIC Quick Reference:

<p><b>Set Servo</b> 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><b>Start Motor</b> 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><b>Stop Motor</b> 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><b>Wait</b> Wait an amount of time measured in seconds. Only use whole numbers on the VEX PIC. <i>Default time: 1.</i></p>	<pre>startMotor(); wait(); stopMotor();</pre>	<pre>startMotor(port8, 63); wait(2.7); stopMotor(port8);</pre>
<p><b>Wait in Milliseconds</b> 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><b>Until Touch</b> The robot waits for the Touch Sensor to be pressed. <i>Default sensor port: NONE.</i></p>	<pre>startMotor(); untilTouch(in10); stopMotor();</pre>	<pre>startMotor(port8, 63); untilTouch(in10); stopMotor(port8);</pre>
<p><b>Until Release</b> The robot waits for the Touch Sensor to be released. <i>Default sensor port: NONE.</i></p>	<pre>startMotor(); untilRelease(in10); stopMotor();</pre>	<pre>startMotor(port8, 63); untilRelease(in10); stopMotor(port8);</pre>
<p><b>Until Bump</b> The robot waits for the Touch Sensor to be pressed in and then released out. <i>Default sensor port: NONE.</i></p>	<pre>startMotor(); untilBump(in10); stopMotor();</pre>	<pre>startMotor(port8, 63); untilBump(in10); stopMotor(port8);</pre>
<p><b>Until Button Press</b> 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><b>Until Sonar - Less Than</b> The robot waits for the Sonar Sensor to read a value in cm less than the threshold. <i>Default threshold and sensor port: NONE.</i></p>	<pre>startMotor(); untilSonarLessThan(45, in2); stopMotor();</pre>	<pre>startMotor(port8, 63); untilSonarLessThan(45, in2); stopMotor(port8);</pre>
<p><b>Until Sonar - Greater Than</b> The robot waits for the Sonar Sensor to read a value in cm greater than the threshold. <i>Default threshold and sensor port: NONE.</i></p>	<pre>startMotor(); untilSonarGreaterThan(45, in2); stopMotor();</pre>	<pre>startMotor(port8, 63); untilSonarGreaterThan(45, in2); stopMotor(port8);</pre>

# ROBOTC Natural Language - VEX PIC Quick Reference:

<p><b>Until Potentiometer - Greater Than</b> The robot waits for the Potentiometer Sensor to read a value greater than a set position. <i>Default threshold and sensor port: NONE.</i></p>	<pre>startMotor(); untilPotentiometerGreaterThan(     900, in4); stopMotor();</pre>	<pre>startMotor(port8, 63); untilSonarGreaterThan(900, in4); stopMotor(port8);</pre>
<p><b>Until Potentiometer - Less Than</b> The robot waits for the Potentiometer Sensor to read a value less than a set position. <i>Default threshold and sensor port: NONE.</i></p>	<pre>startMotor(); untilPotentiometerLessThan(     500, in4); stopMotor();</pre>	<pre>startMotor(port8, 63); untilSonarLessThan(40, in4); stopMotor(port8);</pre>
<p><b>Until Dark</b> The robot waits for the Light Sensor to read a value greater than the threshold. <i>Default threshold and sensor port: NONE.</i></p>	<pre>startMotor(); untilDark(400, in4); stopMotor();</pre>	<pre>startMotor(port8, 63); untilDark(400, in4); stopMotor(port8);</pre>
<p><b>Until Light</b> The robot waits for the Light Sensor to read a value less than the threshold. <i>Default threshold and sensor port: NONE.</i></p>	<pre>startMotor(); untilLight(400 in4); stopMotor();</pre>	<pre>startMotor(port8, 63); untilLight(400, in4); stopMotor(port8);</pre>
<p><b>Until Rotations</b> The robot waits for an encoder to reach a specified whole number of rotations. <i>Default rotations, encoder: NONE</i></p>	<pre>startMotor(); untilRotations(2, in3); stopMotor();</pre>	<pre>startMotor(port8, 63); untilRotations(2, in3); stopMotor(port8);</pre>
<p><b>Until Encoder Counts</b> The robot waits for an encoder to reach a specified number of encoder counts. <i>Default counts, encoder: NONE.</i></p>	<pre>startMotor(); untilEncoderCounts(990,     in3); stopMotor();</pre>	<pre>startMotor(port8, 63); untilEncoderCounts(990, in3); stopMotor(port8);</pre>
<p><b>LED ON</b> Turn an LED in a specified digital port ON. <i>Default sensor port: in12.</i></p>	<pre>turnLEDOn(); wait(); turnLEDOff();</pre>	<pre>turnLEDOn(dgt17); wait(0.5); turnLEDOff(dgt17);</pre>
<p><b>LED OFF</b> Turn an LED in a specified digital port OFF. <i>Default sensor port: in12.</i></p>	<pre>turnLEDOn(); wait(); turnLEDOff();</pre>	<pre>turnLEDOn(dgt17); wait(0.5); turnLEDOff(dgt17);</pre>
<p><b>Flashlight ON</b> 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><b>Flashlight OFF</b> 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><b>Robot Type</b> Choose which robot you are using (Recbot or Swervebot). <i>Default bot: none.</i></p>	<pre>robotType();</pre>	<pre>robotType(squarebot);</pre>

# ROBOTC Natural Language - VEX PIC Quick Reference:

<b>Forward</b> The robot drives straight forward. <i>Default speed: 95.</i>	<pre>forward(); wait(); stop();</pre>	<pre>forward(63); wait(2.0); stop();</pre>
<b>Backward</b> The robot drives straight backward. <i>Default speed: -95.</i>	<pre>backward(); wait(); stop();</pre>	<pre>backward(63); wait(2.0); stop();</pre>
<b>Point Turn</b> 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>
<b>Swing Turn</b> 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>
<b>Stop</b> The robot halts both driving motors, coming to a stop.	<pre>forward(); wait(); stop();</pre>	<pre>forward(63); wait(2.0); stop();</pre>
<b>Move Straight - for Time</b> The robot will use encoders to maintain a straight path for a specified time in seconds. <i>Default rotations, rightEncoder, leftEncoder: 5, in2, in3 + interrupts.</i>	<pre>moveStraightForTime(7, in5, in3); stop();</pre>	<pre>moveStraightForTime(7, in5, in3); stop();</pre>
<b>Move Straight - for Rotations</b> The robot will use encoders to maintain a straight path for a specified distance in encoder rotations. <i>Default rotations, rightEncoder, leftEncoder: 1, in2, in3 + interrupts.</i>	<pre>moveStraightForRotations( 4, in5, in3); stop();</pre>	<pre>moveStraightForRotations(4, in5, in3); stop();</pre>
<b>Tank Control</b> 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>
<b>Arcade Control</b> 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>