

## Programming Challenge

### Quick Tap Phase 1

#### Challenge Description

Write a program that counts the number of times a bumper or limit switch is pressed over a 15 second interval. Begin by declaring two integer variables named "pressCount" and "finalCount". Every time that the touch sensor is pressed add 1 to the variable "pressCount". At the end of the 15 seconds, assign the value of "pressCount" into the variable named "finalCount".

Keep your robot connected to the computer over VEXnet or USB to observe the value stored in "finalCount" using the "Robot > Debugger Windows > Global Variables" window. (See page 4 for details on using the "Global Variables" window.)

#### Notes

- What are the variables that you will need to add to your program?
- How long should your program count?
- Declare the variables and assign a value of 0 to each of them.
- Each "press" should result in a change of only +1 in the value of the variable "pressCount". (this is going to be problematic at first and so don't be surprised)
- Increment "pressCount" using the code: `pressCount = pressCount + 1;`
- Begin by following pseudocode that describes in detail what you want your program to do.

#### Pseudocode Phase 1

Below you will find a pseudocode description of what you want your program to do.

```

begin program
  declare an integer variable "pressCount"
  declare an integer variable named "finalCount"
  set both integer variables to zero
  clear timer 1
  while(timer is less than 15 seconds)
  {
    if(the limit switch is pressed)
    {
      increment pressCount by one
    }
  }
  set "finalCount" equal to "pressCount"
end program

```

Turn the pseudocode example into ROBOTC code and test it.

## Programming Challenge

### Quick Tap Phase 2

When you worked on Phase 1, you may have noticed that the value of "pressCount" incremented many times each time the limit switch was pressed. This is because "pressCount" is embedded into a loop and the robot counts very fast. Since the robot can count faster than a person can touch, the number stored in "pressCount" grows very large with each tap. This can be solved by waiting for the limit switch to be released before the robot increments "pressCount" a second time. Study the updated Pseudocode below which adds an "idle loop" into the program forcing the robot to wait for the limit switch to be released before the program counts again.

```
task main
{
  declare an integer variable named "pressCount"
  declare an integer variable named "finalCount"
  set both integer variables to zero
  clear timer 1
  while(less than 15 seconds have elapsed on timer 1)
  {
    if(the limit switch is pressed)
    {
      increment pressCount by one
      while(the limit switch is still pressed)
      {
        //waits for limit switch to be released so that...
        //...each press is only counted once
      }
    }
  }
  set "finalCount" equal to "pressCount"
}
```

Turn the pseudocode example into ROBOTC code and test it.

## Programming Challenge

### Quick Tap Phase 3

When you implemented Pseudocode Phase 2, you may have noticed that the robot sometimes counted a press twice. This issue is also caused by the robot counting too fast and as the limit switch is released, a second press is registered by the robot. A brief delay, insignificant to humans (but not the robot), can be added to solve the problem.

```
task main
{
  declare an integer variable named "pressCount"
  declare an integer variable named "finalCount"
  set both integer variables to zero
  clear timer 1
  while(less than 15 seconds have elapsed on timer 1)
  {
    if(the limit switch is pressed)
    {
      increment pressCount by one
      wait for 5 milliseconds

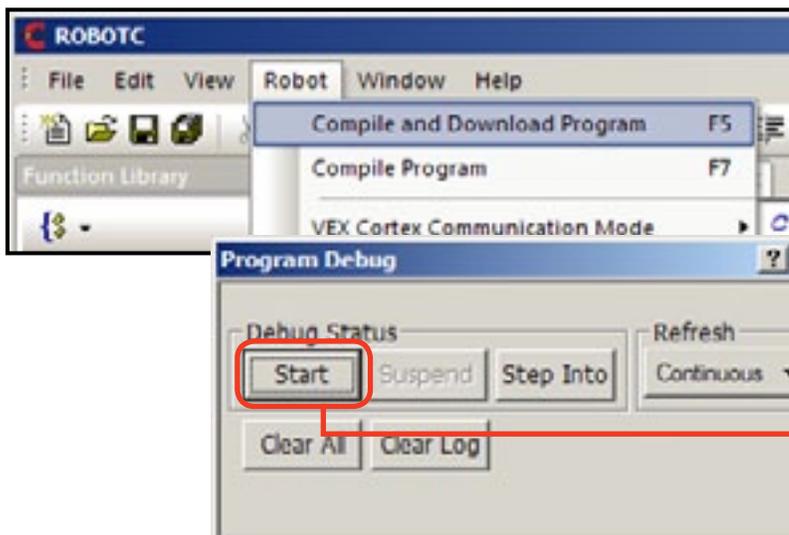
      while(the limit switch is still pressed)
      {
        //waits for limit switch to be released so that...
        //...each press is only counted once
      }
    }
  }
  set "finalCount" equal to "pressCount"
}
```

Turn the pseudocode example into ROBOTC code and test it.

## Programming Challenge

# Debugger Window: Global Variables

The Global Variables window is a powerful part of the ROBOTC debugger. When shown, it displays the current status of any variables declared in the program. Use the following steps to open the Global Variables window and view the information it provides.

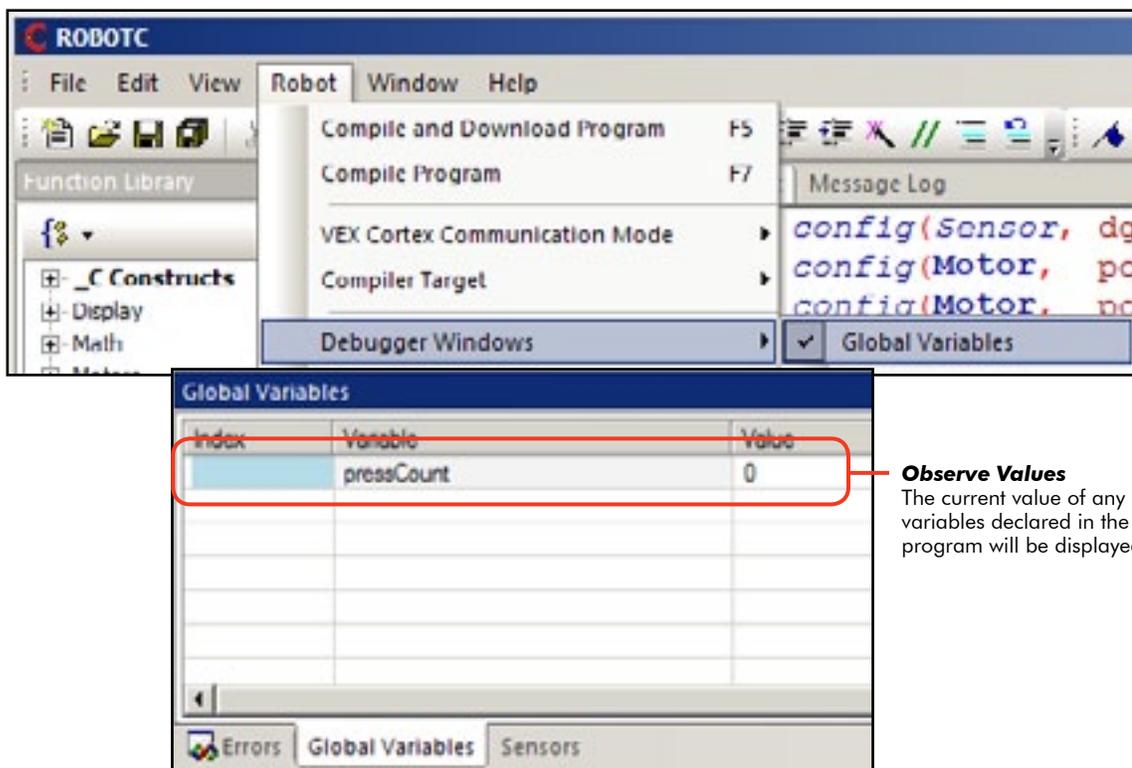


**Robot > Compile and Download Program**  
Compile and Download your program to the robot.

**Program Debug**  
The Program Debug window will open once the download is complete.

**Start Button**  
Press the "Start" button to initialize the program.

**Robot > Debugger Windows > Global Variables**  
Open up the Global Variables window.



**Observe Values**  
The current value of any variables declared in the program will be displayed.