

FRC 2019 BETA

Table of Contents

Alpha Test (Publically available)	4
Alpha Test Info	5
Installing VS Code	6
VS Code Basics and WPILib in VS Code	16
Creating a new WPILib project in VS Code.....	19
Importing an Eclipse project into VS Code	26
2019 Beta Testing - Getting Started (Beta teams only)	29
Welcome	30
Accessing the 2019 Beta Project	31
Reporting Progress	34
Trackers - Reporting Bugs.....	37
Monitoring Via Email Notifications	39
2019 Beta Testing - Tasks	43
2019 Beta Task Overview.....	44
International Team Note.....	46
Task 0: Beta Test Tools and Documentation Familiarization	47
Task 1 - Objective 1: 2019 Software Installation	49
Task 2 - Objective 1: Configure the roboRio	50
Task 2 - Objective 2: Create, Build, and Load Program for Benchtop Testing...	51
Task 3 - Objective 1: Port Existing Robot Code to 2019 System.....	52
(Optional) Task 3 - Objective 3: Offseason Competition with Existing FRC Robot.....	53

FRC 2019 Beta

Task 4 - Objective 1: Testing New Features	54
2019 Beta Specific Documentation (Beta Teams Only).....	55
Installing VS Code for FRC.....	56
VS Code Basics and WPILib in VS Code	60
Creating a new WPILib project in VS Code.....	63
Importing an Eclipse project into VS Code	70
Installing LabVIEW for FRC 2019 (LabVIEW only)	73
Installing the FRC Update Suite (All Languages).....	96
Imaging your roboRIO	117

Alpha Test (Publically available)

Alpha Test Info

As part of our continual improvements to the FRC Control System software, for the 2019 FRC season the current plan is to switch from Eclipse (with built-in Ant scripts as the build/deploy tooling) as the supported IDE for C++\Java development to Microsoft's Visual Studio Code IDE with Gradle (using a community created project, GradleRIO, pulled under the WPILib umbrella) as the build/deploy tooling.

In order to help teams gain familiarity with this new development environment, and to provide a more extensive opportunity to test this change, we are conducting an open Alpha test of the new IDE and tooling with the 2018 libraries. This means that you can follow the instructions in the subsequent articles and get your 2018 robot up and running with the new environment to take it for a test drive.

You can learn more about these changes, and other plans for 2019, in the Building and Contributing to WPILib presentation from the FRC Championship: <http://wp.wpi.edu/wpilib/files/2018/05/Building-and-Contributing-to-WPILib-2018.pdf>

We encourage you to try out this new platform, as well as let other teams know about the changes coming up in the 2019 season. Feature requests and contributions are welcome as always!

Report Bugs/Feature Requests

Bugs and feature requests should be reported using the Issue trackers on the appropriate WPILib Suite repo:

- [Vscode-wpilib](#): For all issues related to the WPILib extension for VS Code (wizards, creating functions, launching tasks, etc.)
- [GradleRIO](#): For issues related specifically to Gradle task such as build/deploy/etc.

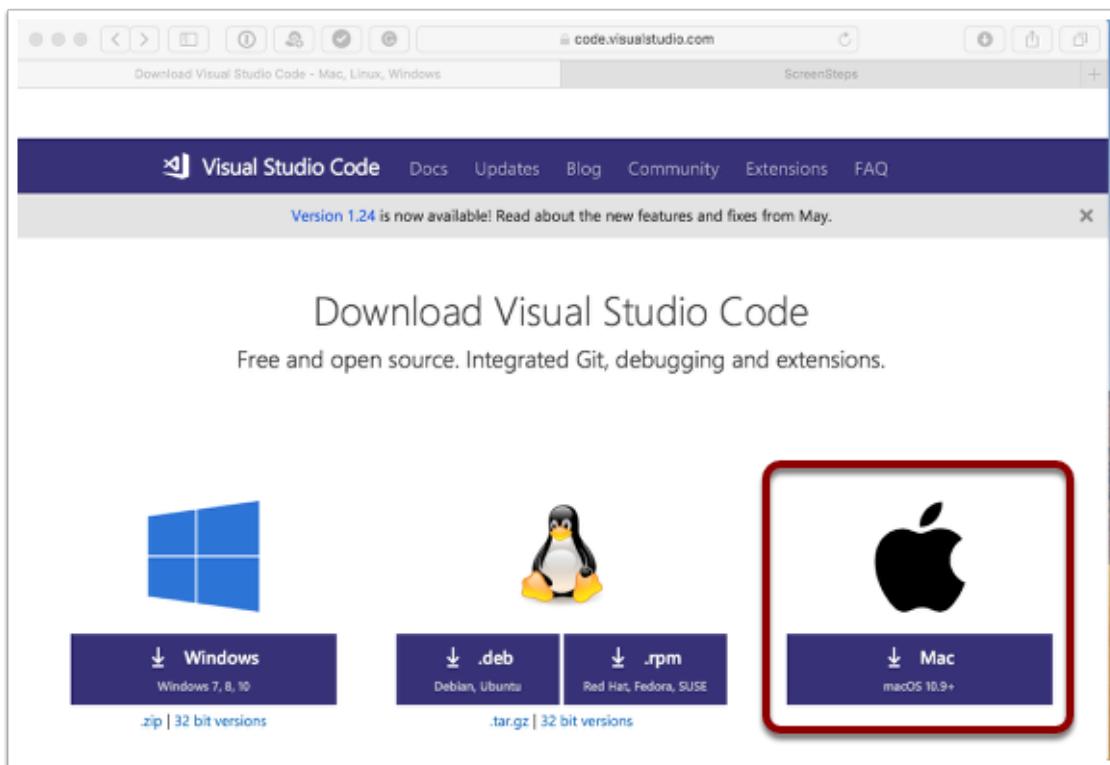
Questions

Questions can be directed to the WPILib Gitter: <https://gitter.im/wpilibsuite/wpilib?source=orgpage>

Installing VS Code

Getting VS Code

Visual Studio Code is a lightweight free to use platform for software development. It supports Java and C++ development and with the supplied plugins you can develop FRC programs. Starting with the 2019 FRC season, VS Code will be the Integrated Development Environment that is supported. To get VS Code for your platform navigate to: <https://code.visualstudio.com/download> and select the version that matches your development OS. In this example, we'll install the Mac version of VS Code.

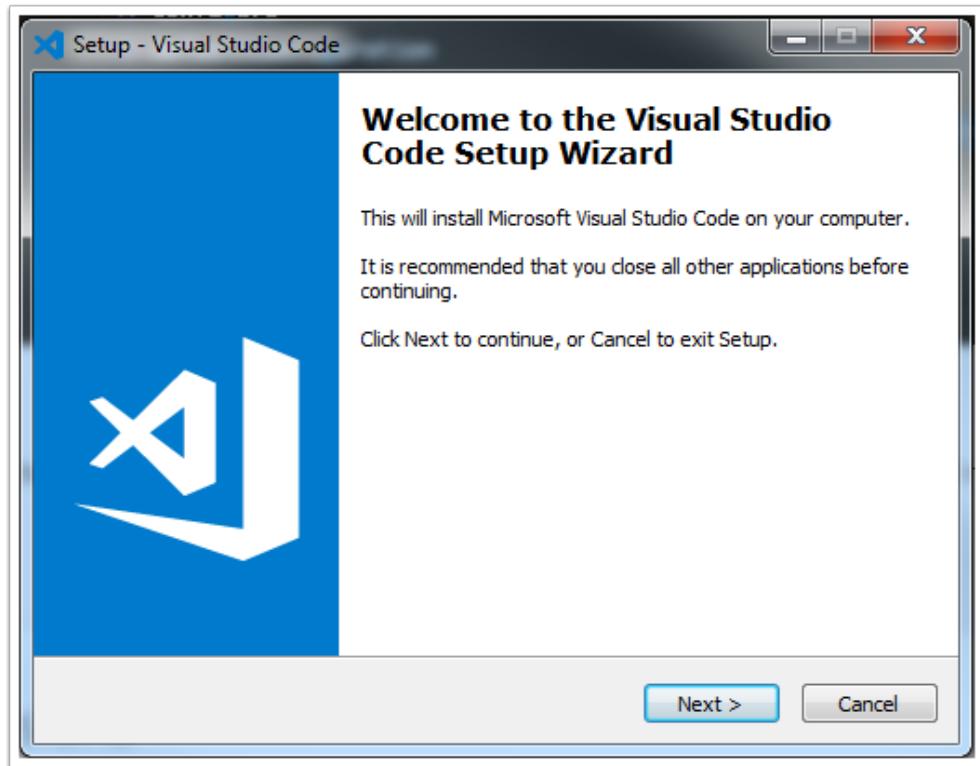


Choose the version for your hardware and install it by following the on-screen instructions which may vary slightly by platform.

⚠ Note: VS Code version 1.25 or later is required to work properly with the WPILib plugin.

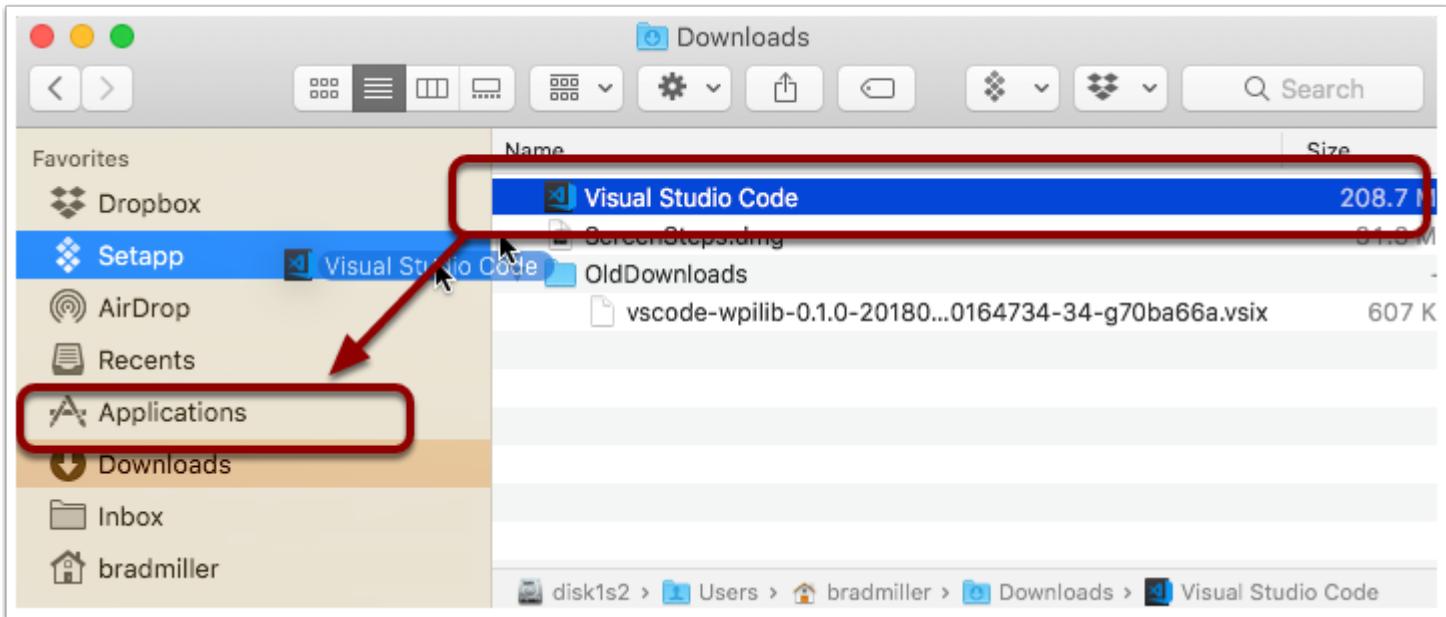
Windows

For Windows, run the downloaded executable and follow the on-screen directions to complete the installation.



Mac

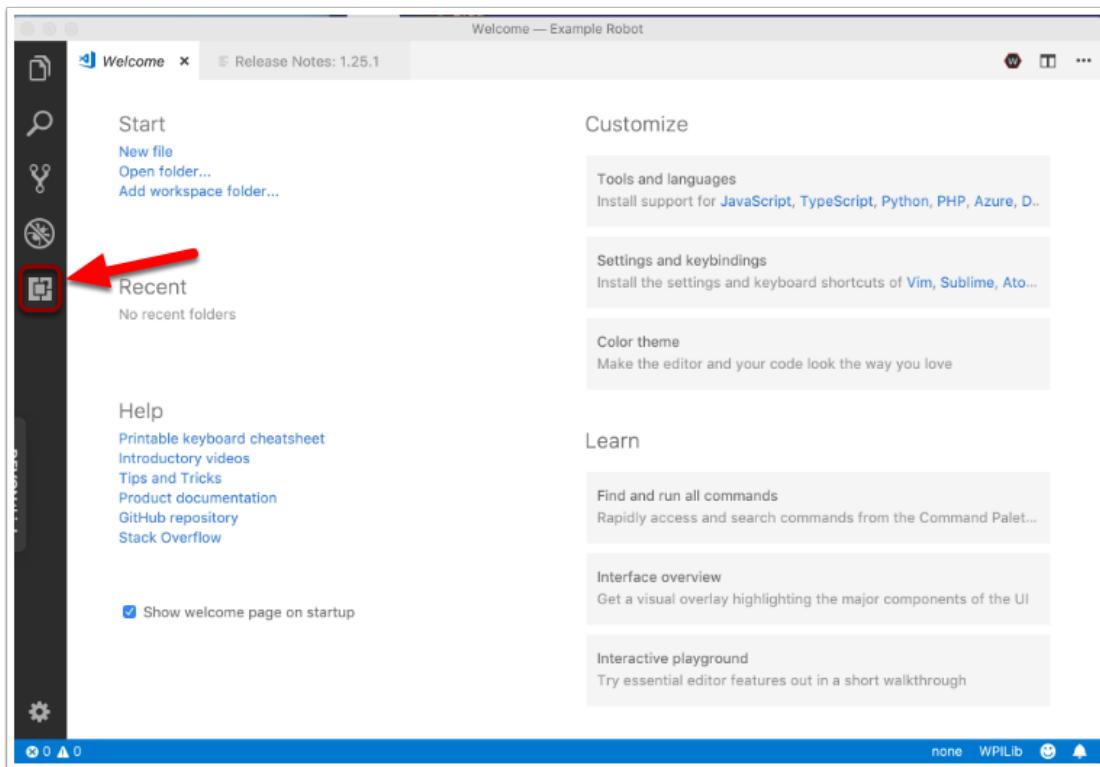
For the Mac version it is installed by copying the executable file to the Applications directory as shown in the example below.



Install some useful plugins

Run VS Code to see the interface. The first time it starts you'll be presented with a Welcome screen. You can browse through it to learn a little about the interface for VS Code. To get up and running with VS Code you need to install extensions for C++, Java, and WPILib. The extensions directory can be loaded by pressing the bottom button in the vertical toolbar as shown.

FRC 2019 Beta



The extensions to install are:

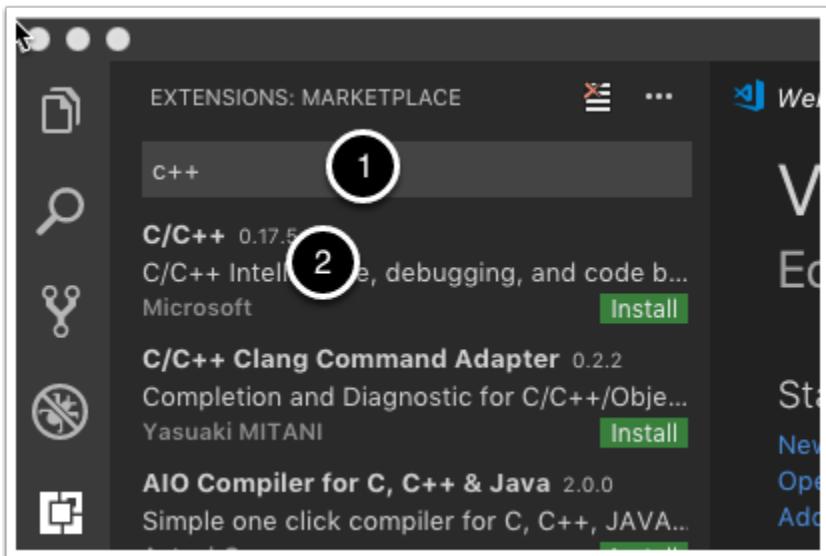
1. For C++ coding - VS Code C++ extension
2. For Java coding - VS Code Java extension pack
3. For all teams - WPIlib vsix file containing the extension

⚠ Note: The WPIlib extension will currently issue spurious warnings if only 1 of the two languages is installed. If you would like to avoid this, install both regardless of coding language. This is a known issue that we are looking into.

Installing Language Extensions

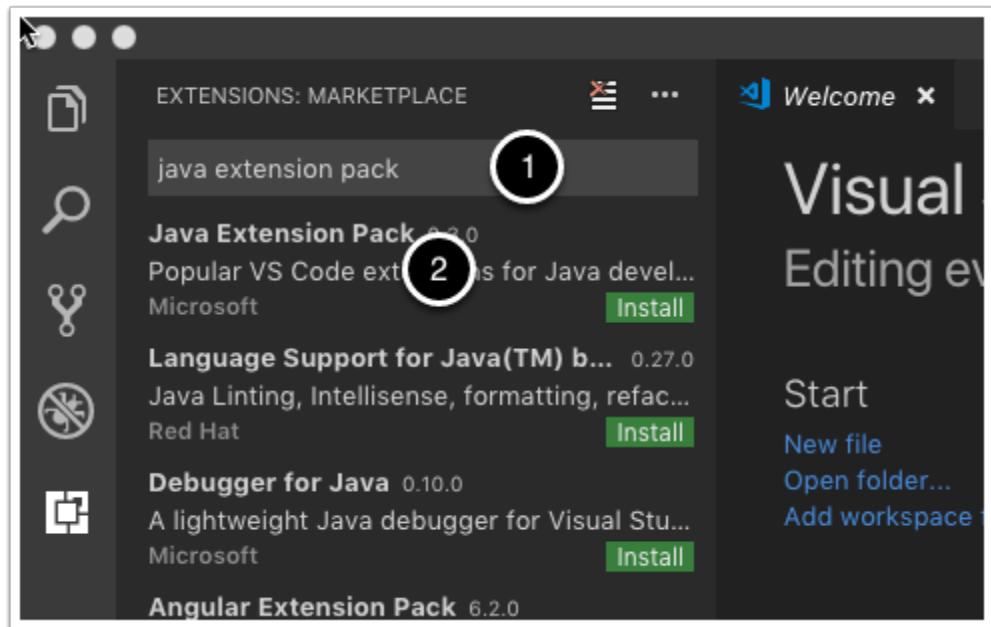
C++

Type the extension name for C++ into the search bar (1) and then (2) select C/C++ and click "install".



Java

Type the extension name for Java into the search window as shown below (1) then select the Java extension pack (2) and click "install".



Download the WPILib VS Code extension

For the Alpha release, the WPILib VS Code extension is not part of the VS Code marketplace and it has to be downloaded and installed manually.

The download can be found at: <https://github.com/wpilibsuite/vscode-wpilib/releases/tag/v2019.0.0-alpha-4>

Click on the latest alpha file and you will be prompted to download the .vsix file.



Note: The Alpha files are intended for the public alpha and use the 2018 libraries and roboRIO image. The Beta files are intended for the closed Beta and require the 2019 roboRIO image.

FRC 2019 Beta

The screenshot shows a GitHub release page for the repository "v2019.0.0-alpha-3". The release is identified by the tag "v2019.0.0-alpha-3" and was released by "ThadHouse" 17 hours ago. The commit hash is "da1f6a8". There is an "Edit" button in the top right corner. Below the release title, there is a section titled "Assets" which lists four items:

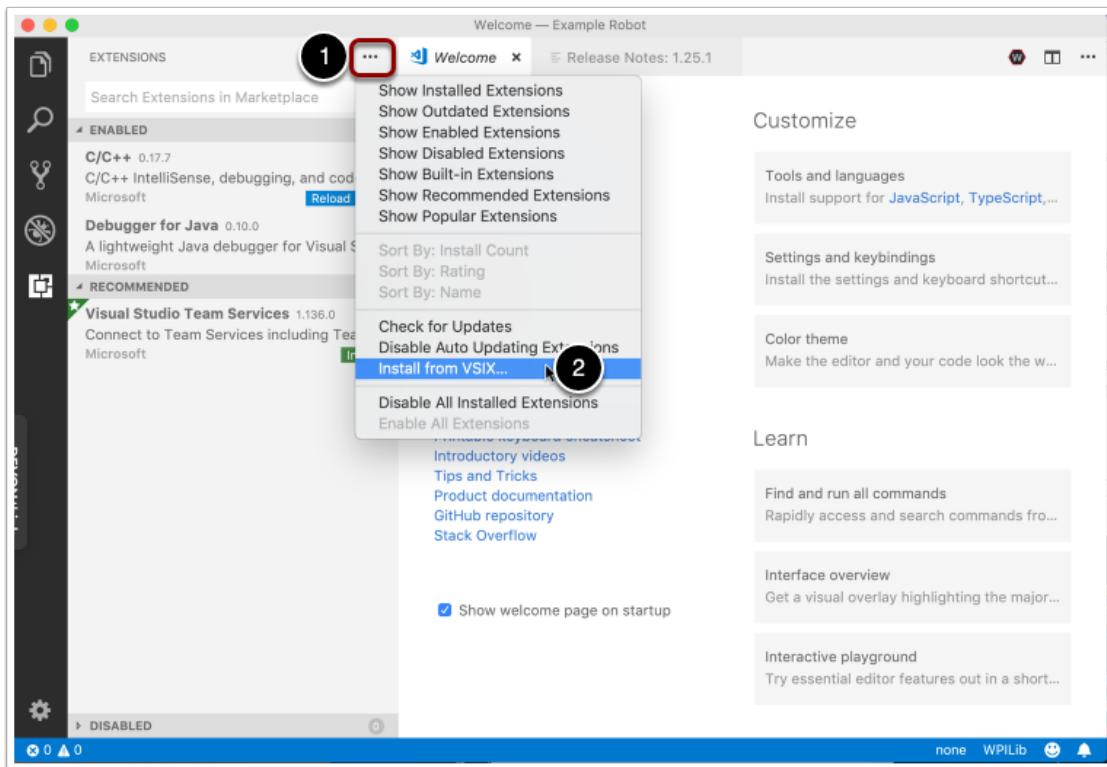
- [vscode-wpilib-2019.0.0-alpha-3.vsix](#) (623 KB) - This item is highlighted with a red border.
- [WPILibStandalone.zip](#) (39.7 MB)
- [Source code \(zip\)](#)
- [Source code \(tar.gz\)](#)

Below the assets, there is a note: "Update readme, changelog and urls in package.json".

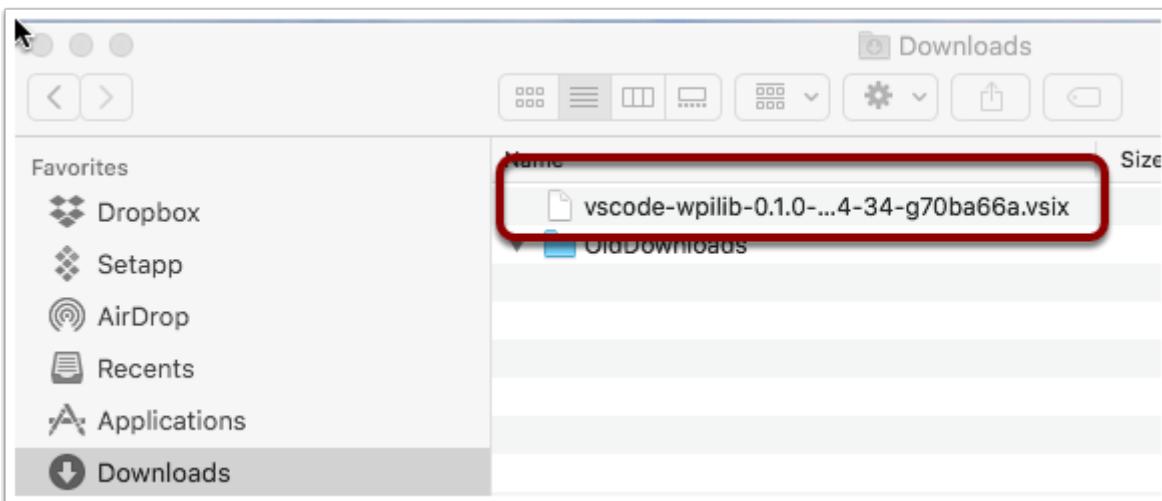
Install the WPILib VSstat Code extension

To install the WPILib extension (1) click on the ... button above the extensions search bar. (2) Select "Install from VSIX...".

FRC 2019 Beta



Select the vsix file the was downloaded in the previous step and click "Install" to install the extension. After the installation finishes, click "Reload Now" to restart VS Code.



C++ Toolchains (C++ Teams Only)

If this installation is on a computer that had a functional Eclipse C++ setup for 2018, you can skip this step for the alpha.

If this is a new machine that was not used for FRC C++ coding in 2018, you will need to install the C++ toolchains.

Installing the C++ Toolchains (C++ teams only)



⚠️ If you have toolchains installed from a prior season, you should uninstall them using Add/Remove and install the new toolchains

FRC 2019 Beta

Download the appropriate C++ Toolchains installer for your platform from <http://first.wpi.edu/FRC/toolchains/>

Windows: Double click on the downloaded file to launch it. If you receive a Security Warning, click Run. Check the box to accept the License Agreement, then click Install. When the install completes, click Finish.

Mac OSX: Double-click on the downloaded file in Finder to unzip it. In Finder, right-click on the "FRC ARM Toolchain.pkg" file, then press the option key on your keyboard, and click "Open". Follow the steps to install the package.

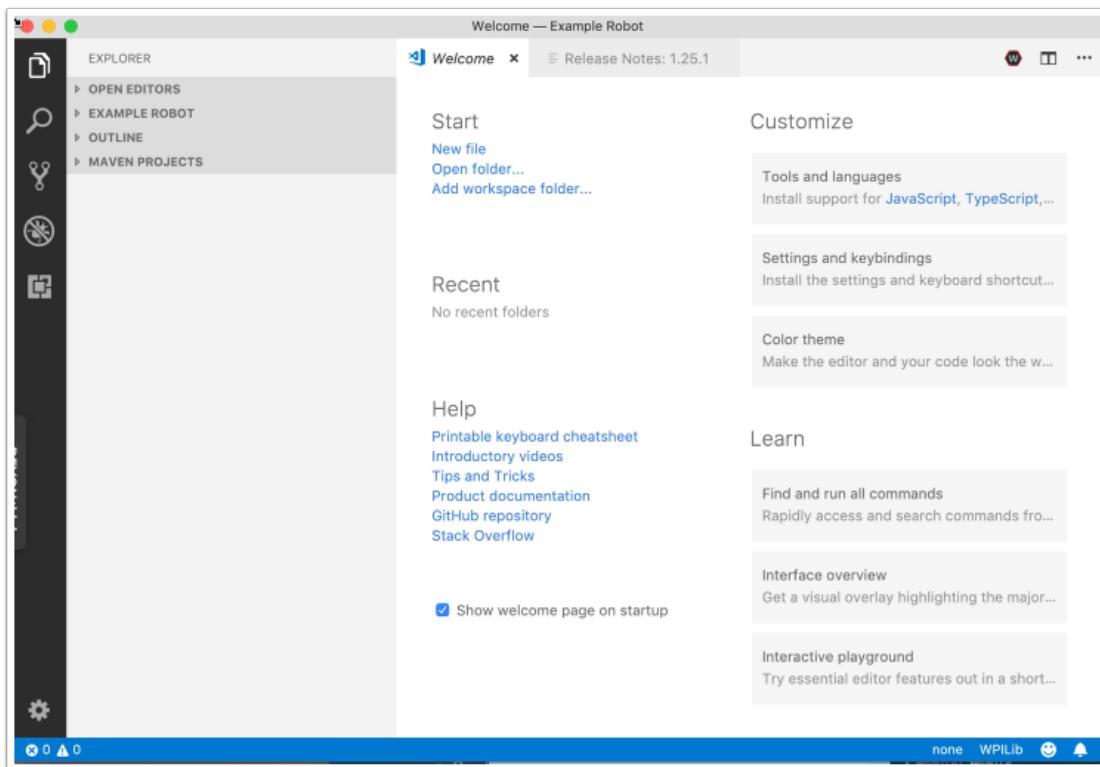
Linux: See the instructions in the text file on the toolchains page.

- ! The Windows toolchains will always install to the root of the system drive.

VS Code Basics and WPILib in VS Code

Microsoft's Visual Studio Code (VS Code) is the new supported IDE for C++ and Java development in FRC, replacing the Eclipse IDE used from 2015-2018. This article introduces some of the basics of using VS Code and the WPILib extension.

VS Code Welcome Page



When VS Code first opens, you are presented with a Welcome page. On this page you will find some quick links that allow you to customize VS Code as well as a number of links to help documents and videos that may help you learn about the basics of VS Code as well as some tips and tricks.

You may also notice a small WPILib logo way up in the top right corner. This is one way to access the features provided by the WPILib extension (discussed further below).

VS Code User Interface

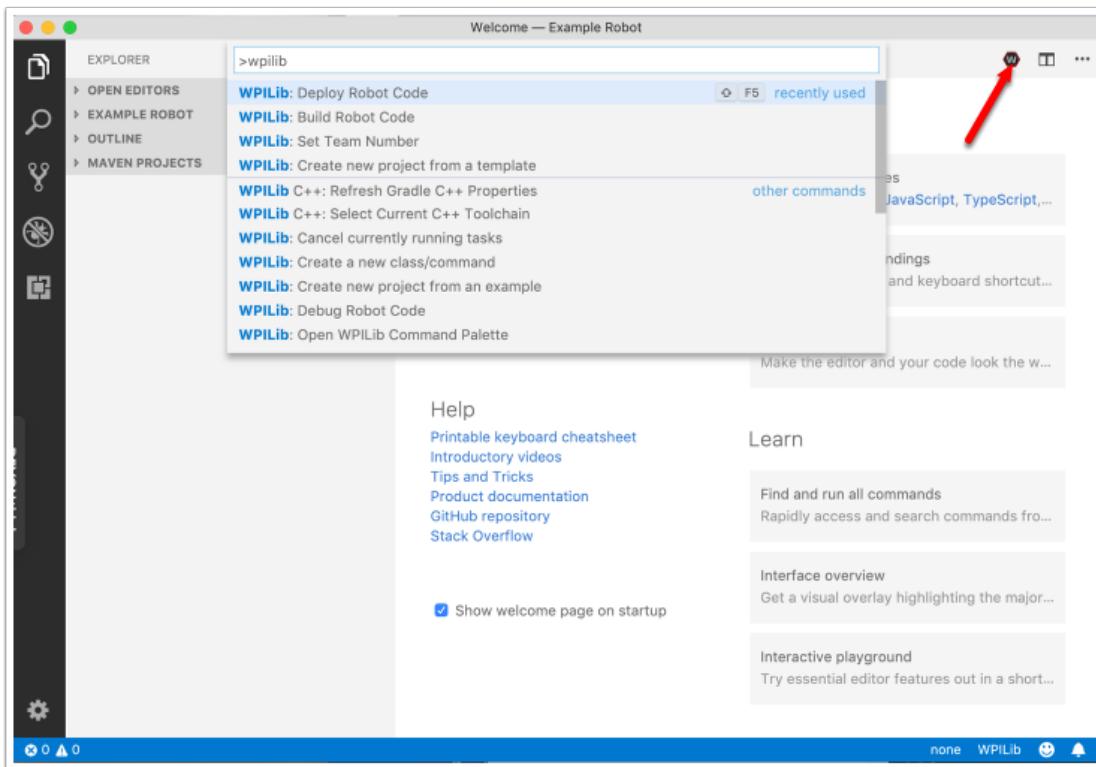
The most important link to take a look at is probably the basic [User Interface document](#). This document describes a lot of the basics of using the VS Code UI and provides the majority of the information you should need to get started using VS Code for FRC.

VS Code Command Palette

The Command Palette can be used to access or run almost any function or feature in VS Code (including those from the WPILib extension). The Command Palette can be accessed from the View menu or by pressing Ctrl+Shift+P (Cmd+Shift+P on Mac). Typing text into the window will dynamically narrow the search to relevant commands and show them in the dropdown.

In the following example "wpilib" is typed into the search box after activating the Command Palette, and it narrows the list to functions containing WPILib.

WPILib Extension



The WPILib extension provides the FRC specific functionality related to creating projects and project components, building and downloading code to the roboRIO and more. You can access the WPILib commands one of two ways:

- By typing "WPILib" into the Command Palette
- By clicking on the WPILib icon in the top right of most windows. This will open the Command Palette with "WPILib" pre-entered

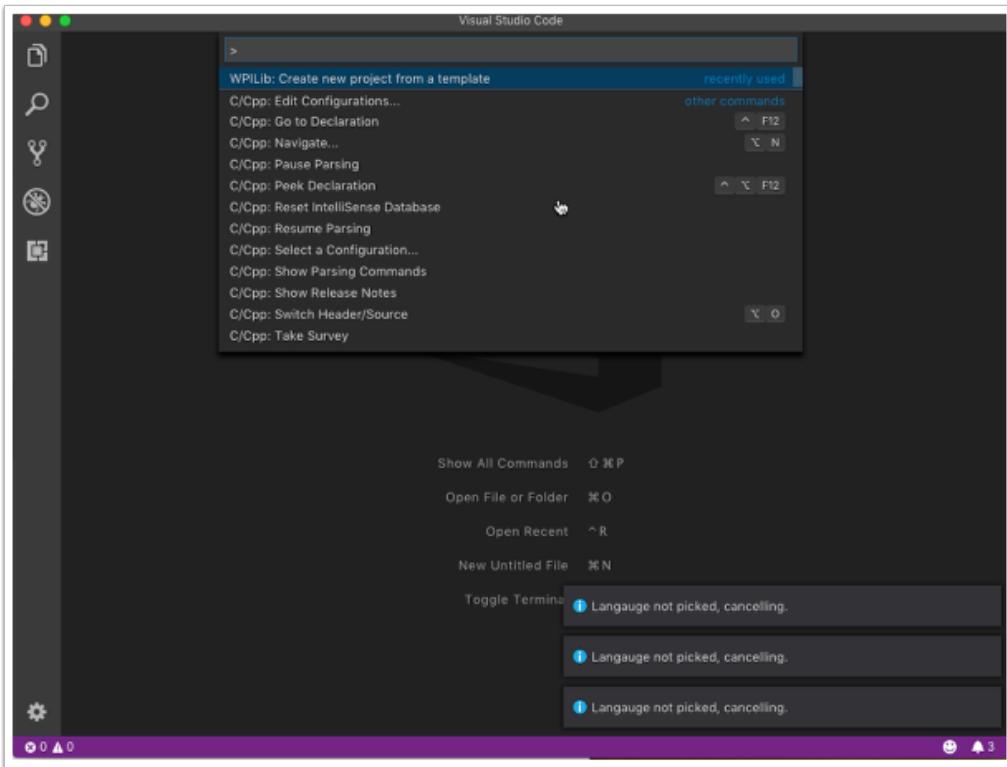
For more information about specific WPILib extension commands, see the other articles in this chapter.

Creating a new WPILib project in VS Code

In this article we will create a new WPILib project in VS Code. In this example we will be making a Command Based Robot, however the same methods apply to creating a project from any of the existing templates or examples.

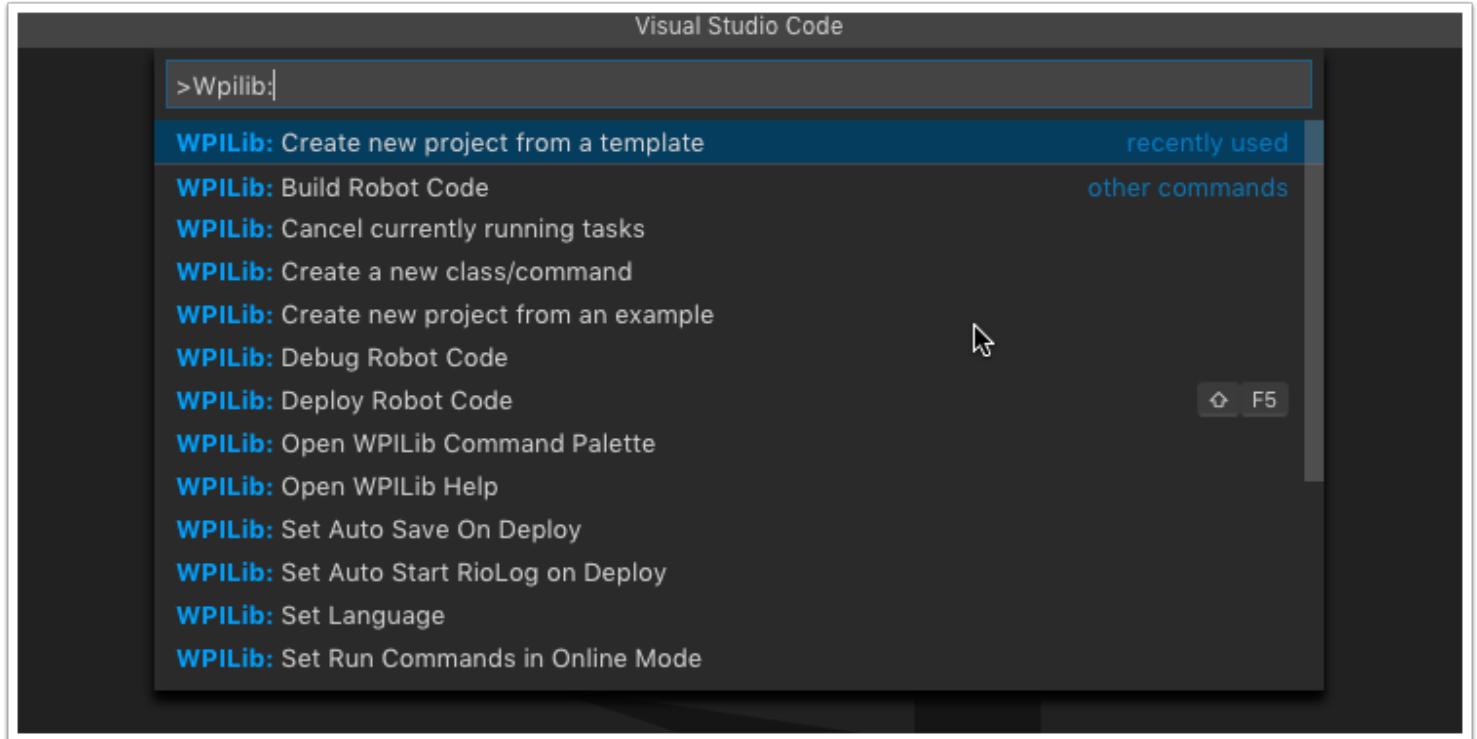
Accessing The Command Palette

Clicking Ctrl+Shift+P will open the command palette. The command palette contains the WPILib commands for creating and interacting with projects.



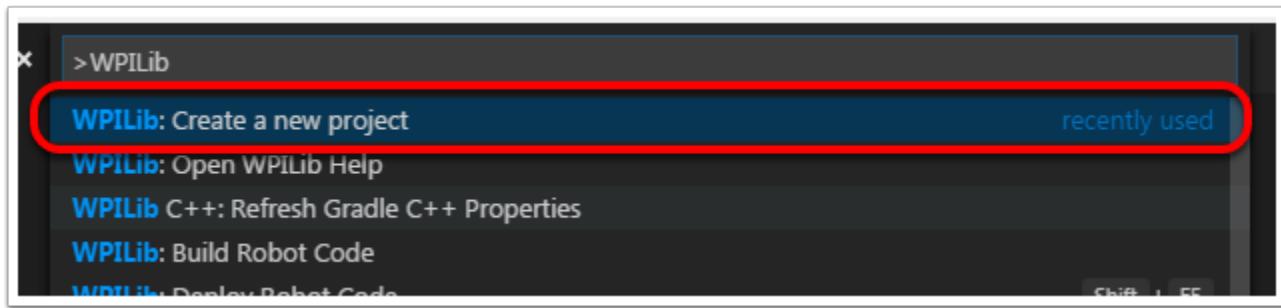
Accessing The WPILib Commands

All WPILib commands start with "WPILib:", so in order to access the WPILib commands type "WPILib:" into the command palette search bar.

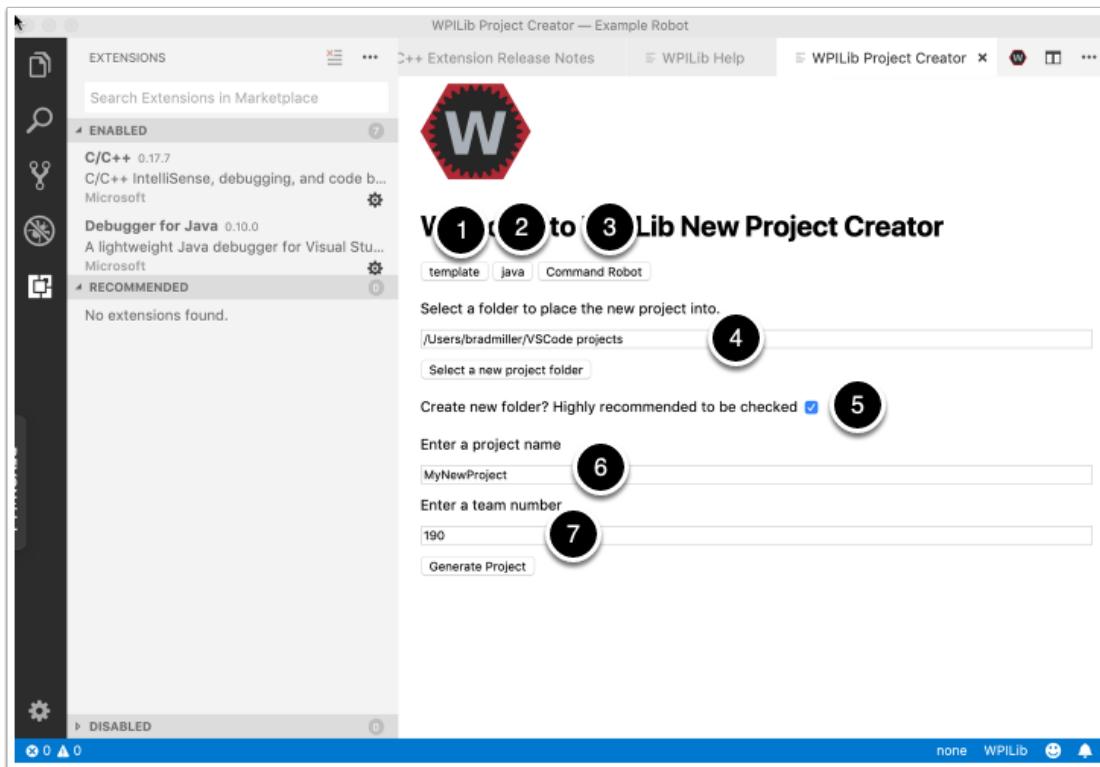


Creating A New WPILib Project

In order to create a new project select the "Create a new project" command. This will show a form with a number of fields where you enter the information required to create your new project.



New project creator window



The steps to create the new project are outlined here:

1. Select the kind of project you want to create. It can be an example project or one of the template projects provided by WPIlib.
2. Select the language that you are using for your project.
3. In the case of a template - select the template type (Timed robot, Iterative robot, Command robot, etc.)
4. Select the folder to place the project.
5. If the "Create new folder" checkbox is checked, a new folder named with the project name is created in the supplied folder. If the checkbox is NOT checked, then the folder supplied is assumed to be empty (will give an error if not) and the project files will be placed into that directory.
6. The project name is used in the project and also to optionally create the folder to place it if the checkbox from the previous step is checked.
7. The team number for the project. This will be used for package names and to locate your robot when deploying code.

FRC 2019 Beta

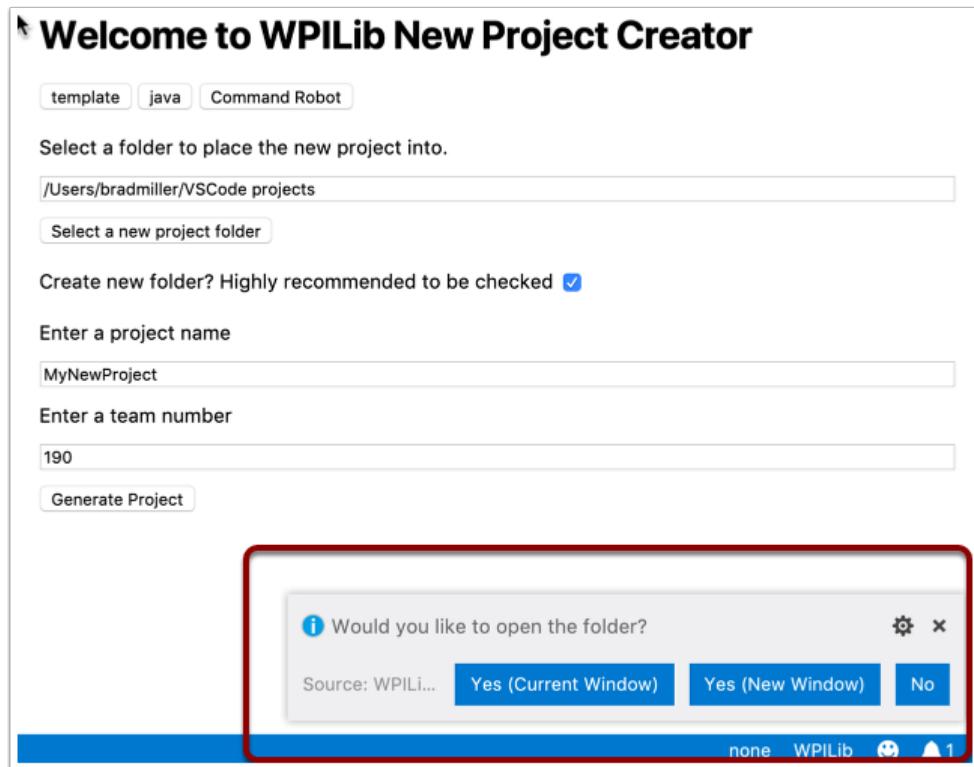
And last, click "Generate Project" and VS Code will create the project in the location specified.

Click the "Select a project type" button and then select your desired type from the dropdown.

- i** Note: If there are any errors generating the project (such as trying to use a non-empty folder with the checkbox unchecked), they will pop up in the bottom right corner of the screen.

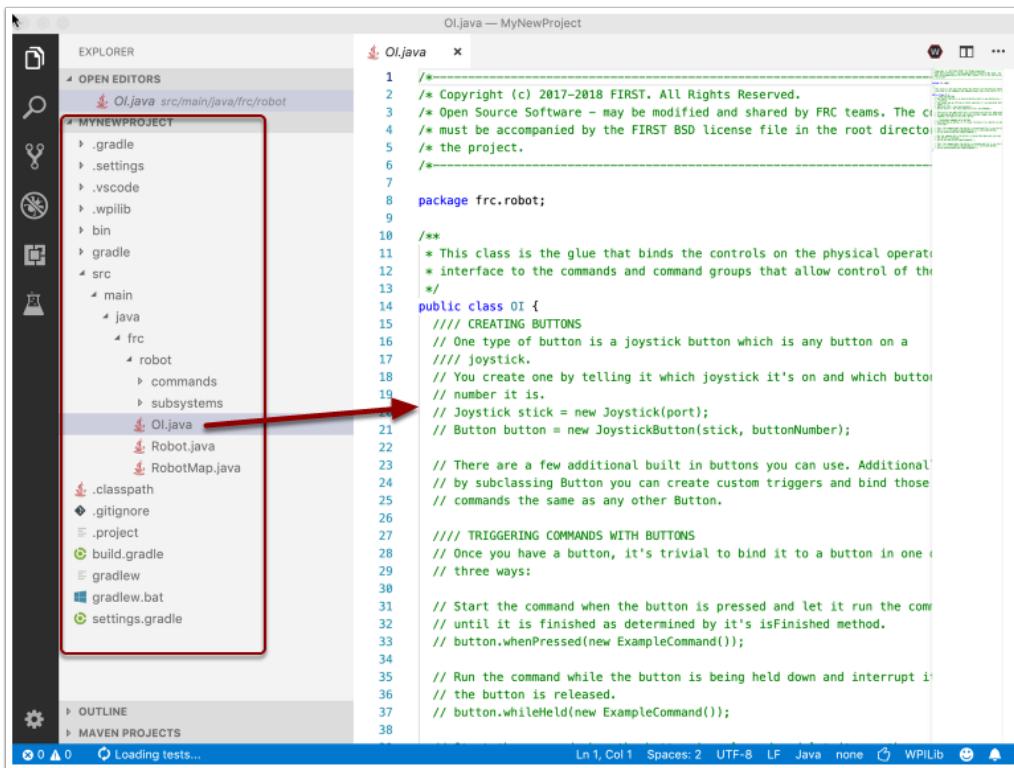
Opening The New Project

After successfully creating your project, VS Code will give you the option of opening the project as shown below. You can choose to do that now or later by typing Ctrl-O (Command+O on mac) and select the folder where you saved your project.

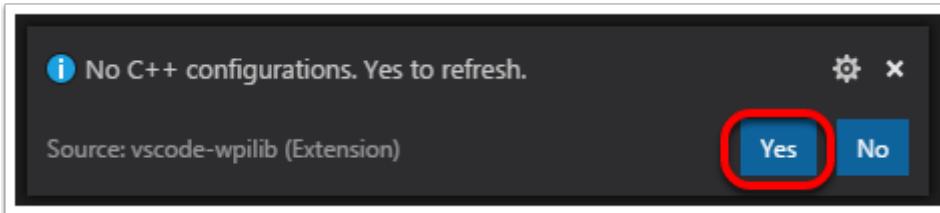


FRC 2019 Beta

Once opened you will see the project hierarchy on the left. Double clicking on the file will open that file in the editor.



C++ Configurations (C++ Only)



For C++ projects, there is one more step to set up IntelliSense. Whenever you open a project, you should get a pop-up in the bottom right corner asking to refresh C++ configurations, click Yes to setup IntelliSense.

Building and Deploying Robot Code

To build the robot project, do one of:

1. Open the Command Palette and select "Build Robot Code"
2. Open the shortcut menu indicated by the ellipses in the top right corner of the VS Code window and select "Build Robot Code"
3. Right-click on the build.gradle file in the project hierarchy and select "Build Robot Code"

The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows the project structure for "MYNEWPROJECT". The "build.gradle" file is selected and highlighted with a red box.
- EDITOR:** Displays the contents of the "build.gradle" file:

```
build.gradle — MyNewProject
plugins {
    id "java"
    id "edu.wpi.first.GradleRIO" version "2019.0.0-alpha-3"
}

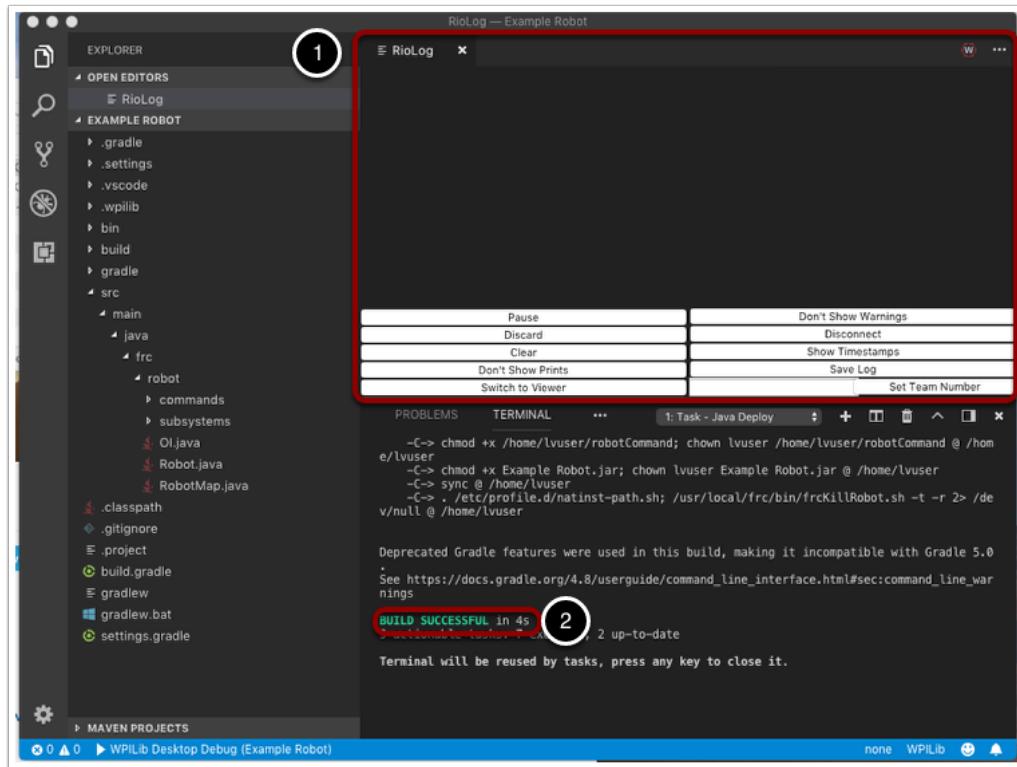
def ROBOT_CLASS = "frc.robot.Robot"

// Define my targets (RoboRIO) and artifacts (deployable files)
// This is added by GradleRIO's backing project EmbeddedTools.
deploy {
    targets {
        target("roborio", edu.wpi.first.gradlerio.frc.RoboRIO) {
            // Team number is loaded either from the .wpilib/wpilib_pref
            // or from command line. If not found an exception will be t
            // You can use getTeamOrDefault(team) instead of getTeamNumbe
            // want to store a team number in this file.
            team = getTeamNumber()
        }
    }
}
```

- OUTPUT:** Shows the build log:
BUILD SUCCESSFUL in 6s
2 actionable tasks: 2 executed
- TERMINAL:** Shows the command being run:
> Executing task: ./gradlew assemble <
- STATUS BAR:** Shows the message "BUILD SUCCESSFUL in 6s" and "2 actionable tasks: 2 up-to-date".
- TOP BAR:** Shows the file name "build.gradle" and the status "Task - Java Build".

Deploy robot code by selecting "Deploy Robot Code" from any of the three locations from the previous instructions. That will build (if necessary) and deploy the robot program to the roboRIO. If successful, you see a "Build Successful" message (2) and the RioLog will open with the console output from the robot program as it runs.

FRC 2019 Beta

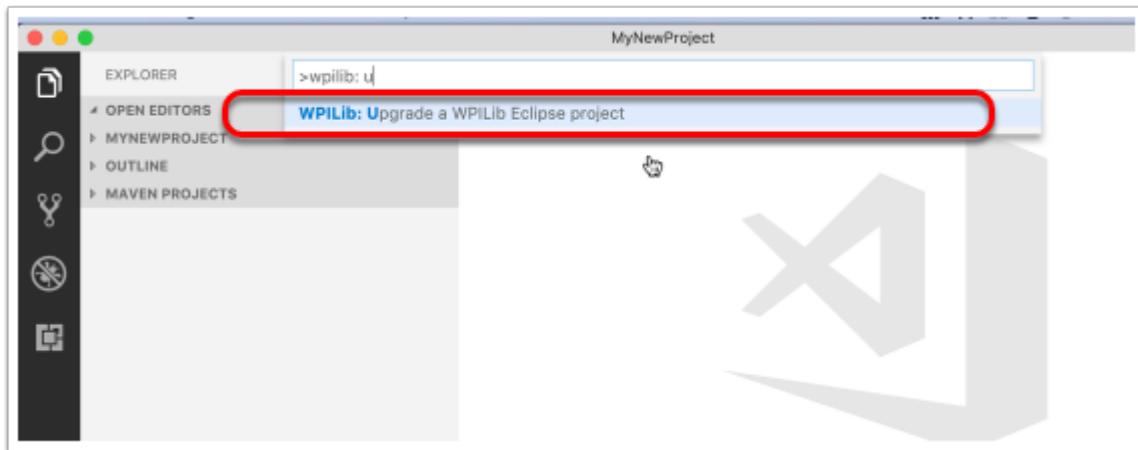


Importing an Eclipse project into VS Code

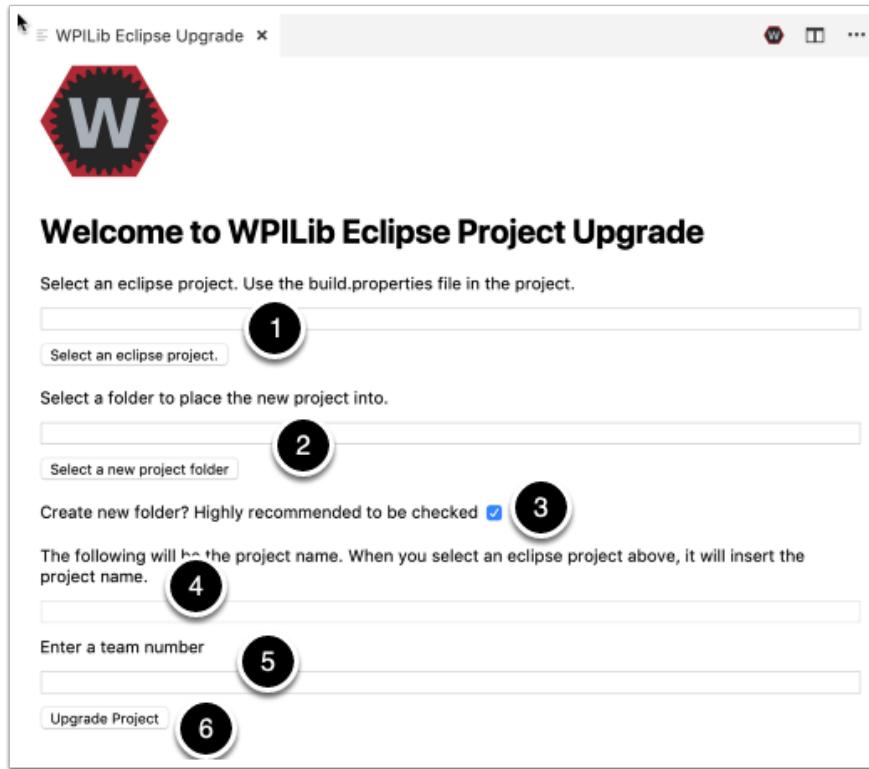
To make it easy for teams to use existing projects with the new IDE, we have implemented a wizard for importing Eclipse projects into VS Code. This will generate the necessary Gradle components and load the project into VS Code.

Launch the Import Wizard

Press Ctrl+Shift+P and type "WPILib" or click the WPILib icon to locate the WPILib commands. Select "Upgrade a WPILib Eclipse Project.



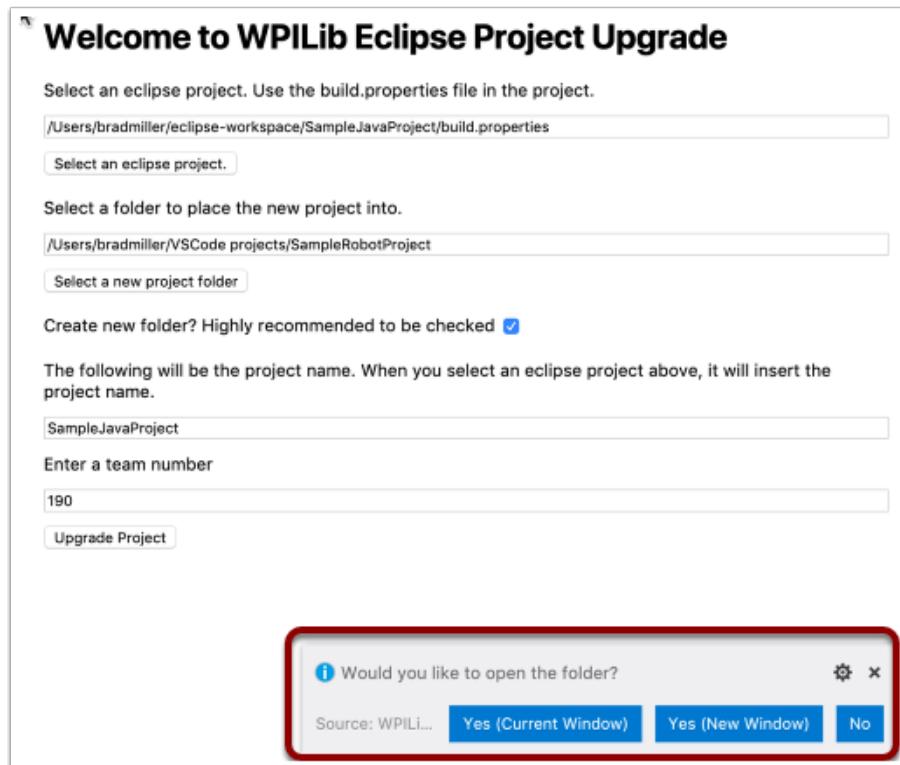
You'll be presented with the WPILib Eclipse Project Upgrade window. This is similar to the process of creating a new project and the window and the steps are shown below.



Perform the following steps to fill in the Eclipse Project Upgrade window:

1. Select the eclipse project to convert. Select the build.properties file in the root directory of the eclipse project.
2. Fill in the new project folder by pressing the "Select a new project folder" button.
3. If the "Create new folder" checkbox is checked, then the project will be stored in a new folder under the one selected in 2. If it is not checked, then the project will be placed in the folder specified. It must be empty in that case.
4. Enter the name of the new project.
5. Enter the team number for the creation of the project and for the robot deployment.
6. And finally, click "Upgrade Project" to begin the upgrade.

The eclipse project will be upgraded and copied into the new project directory from step 3 above. You can then either open the new project immediately or open it later using the Ctrl-O (or Command-O for Mac) shortcut.



C++ Configurations (C++ Only)

For C++ projects, there is one more step to set up IntelliSense. Whenever you open a project, you should get a pop-up in the bottom right corner asking to refresh C++ configurations, click Yes to setup IntelliSense.

2019 Beta Testing - Getting Started (Beta teams only)

Welcome

Thank you for participating in the 2019 Beta testing program.

Before getting started with the [testing tasks](#) please review the following articles.

- [Accessing the 2019 Beta Project](#)
- [Reporting Progress](#)
- [Trackers - Reporting Bugs](#)
- [Monitoring Via Email Notifications](#)
- [2019 Beta Task Overview](#)

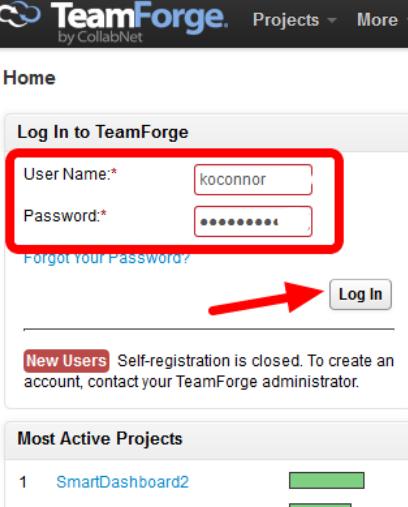
Please use the TeamForge project as your primary means of communication. Members of the control system development team at CTRE, FIRST, NI, and WPI are all monitoring the project and will strive to provide timely feedback and updates.

Accessing the 2019 Beta Project

Note: Only accepted Beta Test teams will have access to the Beta project.

The members identified by your team to communicate with FIRST technical staff will have access to the 2019 Beta Test project on the usfirst.collab.net site.

Signing in



The screenshot shows the TeamForge login interface. At the top, there's a navigation bar with the TeamForge logo and links for 'Projects' and 'More'. Below it, the main area is titled 'Home'. It features a 'Log In to TeamForge' form with 'User Name:' and 'Password:' fields, both of which are highlighted with a red box. A red arrow points from the bottom left towards the 'Log In' button. Below the form, a message says 'New Users' self-registration is closed. To create an account, contact your TeamForge administrator. To the right of the login form is a large 'Welcome to CollabNet TeamForge.' banner with a blue arrow pointing down. Further right is a 'TeamForge Activity' chart showing a line graph of activity over time with two data series: 'Registered Users' (red line with squares) and 'Hosted Projects' (blue line with circles). The chart shows an upward trend for both metrics.

Open your web browser and browse to usfirst.collab.net. Near the top left corner of the page enter the username and password provided in the Beta test email you received, then click "Log In".

Opening the 2019 Beta Project

My Workspace

My Projects All Projects **Project Groups** Project Templates

Page 1 of 2 (25 Items)

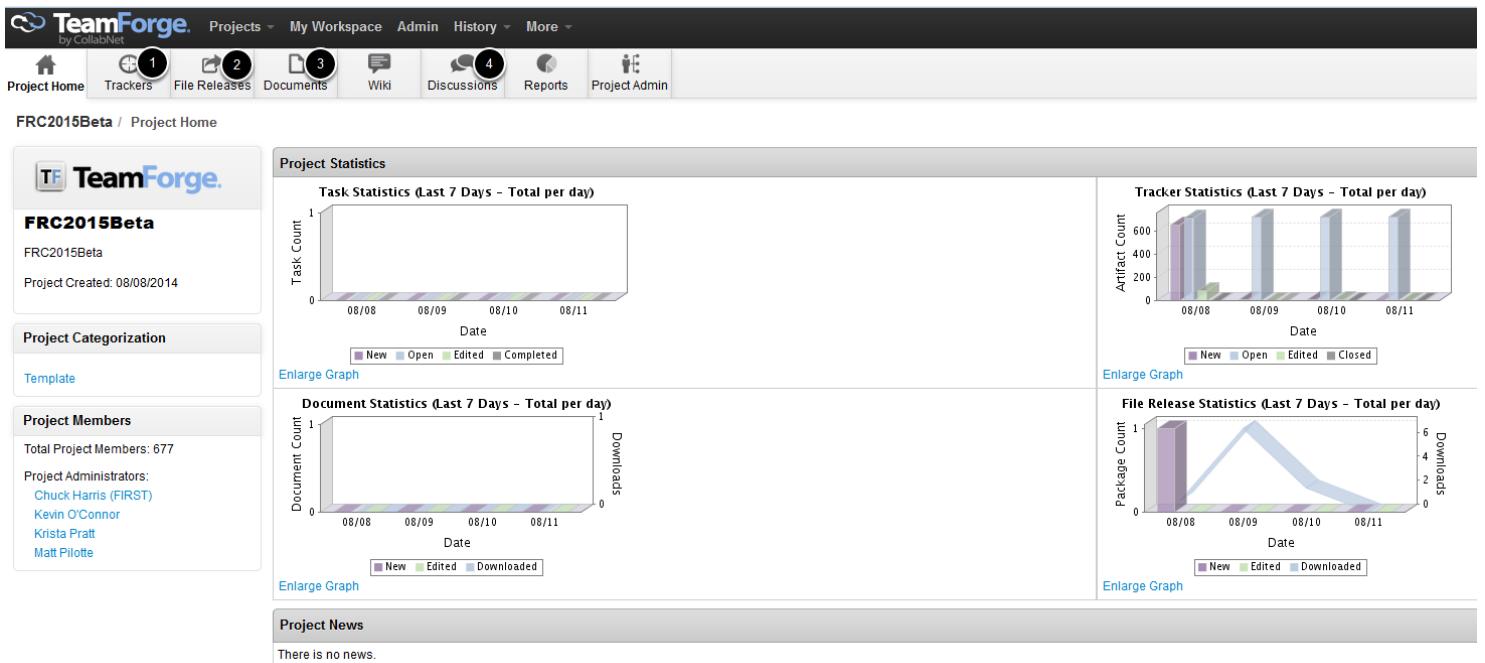
	Project Name	Parent Project	Project Description
<input type="checkbox"/>	adhenning-test-project		Testing various features.
<input type="checkbox"/>	Azalea		FRC Control System Development
<input type="checkbox"/>	CollabNet Agile Baseline 1.5		A template containing a variety of defa Agile development.
<input type="checkbox"/>	FIRST Control System Open Source Policies and Tools		This project contains the policies use current source code. In addition discu
<input type="checkbox"/>	FRC2014Beta		FRC2014Beta
<input type="checkbox"/>	FRC2014BetaPrototype		FRC2014BetaPrototype
<input type="checkbox"/>	FRC2015Alpha		FRC2015Alpha
<input type="checkbox"/>	FRC2015Beta		FRC2015Beta

Your user account should already be added to the 2018 Beta Project. To view the project:

1. Click the Projects tab on the top navigation ribbon.
2. Then click on FRC2019Beta in the projects list (image of 2015 as an example).

FRC 2019 Beta

The Project Homepage



The project homepage contains some statistics and Project News. Throughout the Beta, the Project News section may be updated with the latest information from the FIRST technical staff. The top ribbon contains tabs allowing you to navigate to the different sections of the project:

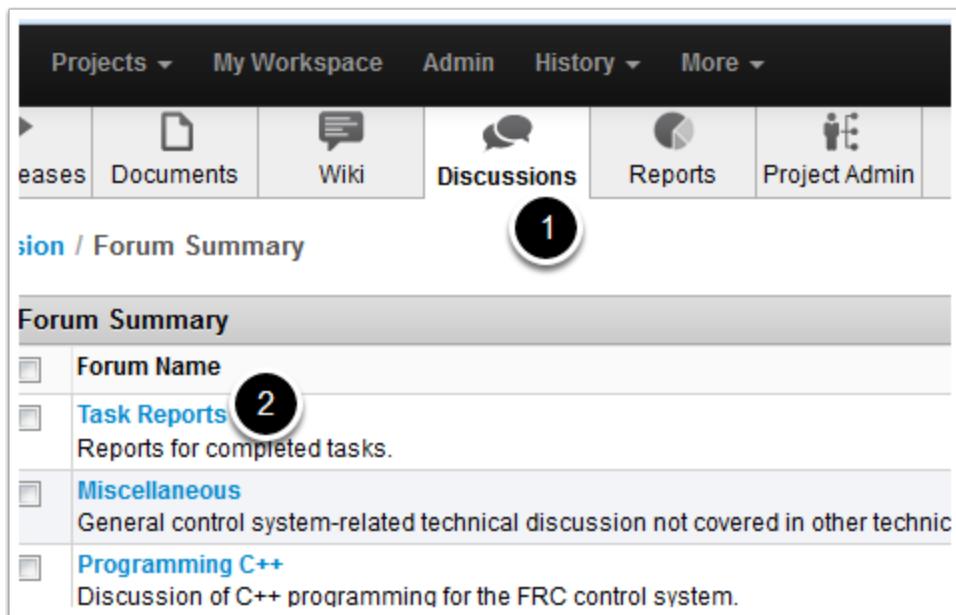
1. Trackers - The Trackers tab is where the bug tracker for the project is located.
2. File Releases - The File Releases tab will host any files you will need to download as part of the Beta test process such as NI Update files.
3. Documents - the Documents tab will contain any documents to be distributed to Beta teams. Much of the documentation for the Beta test will be located on Screensteps but there may still be documents posted here as well.
4. Discussions - The Discussions tab contains a forum which will allow teams to post questions or discussions about the Beta test, software or documentation. Task reports will also be posted here.

Reporting Progress

For 2019 we are going back to posting Task reports in the Discussion section of the Teamforge project. There will be a topic for each task and reports will be posted as replies to that topic. We hope that this will make it easier for teams to see what other teams have been doing and any CS Team replies to other team's reports. Another goal is to hopefully reduce confusion about where or how to post reports and manage the other fields that were present in the old tracker system.

⚠ In order to help us track team participation, please post your team number as part of your task reports.

Accessing the Task Reports Section



The screenshot shows the Teamforge web interface. At the top, there is a navigation bar with links: Projects, My Workspace, Admin, History, More, and several icons for Issues, Documents, Wiki, Discussions, Reports, and Project Admin. A circular badge with the number '1' is positioned over the Discussions icon. Below the navigation bar, the URL 'frc2019 / Forum Summary' is visible. The main content area is titled 'Forum Summary' and contains four forum entries:

- Forum Name**
- Task Reports** (highlighted with a circular badge containing '2')
Reports for completed tasks.
- Miscellaneous**
General control system-related technical discussion not covered in other technical forums.
- Programming C++**
Discussion of C++ programming for the FRC control system.

FRC 2019 Beta

To access the Task Reports section on Teamforge, click on the Discussions tab, then click on the Task Reports section.

The Task Reports Section

The screenshot shows the 'Task Reports - Topic Summary' page. At the top, there is a message: 'To receive email from this mailing list, monitor this discussion forum.' Below this, there are details for the topic: Name: Task Reports, Description: Reports for completed tasks, Type: Public, and Mailing List: taskreports-team_test_template@usfirst.collab.net. To the right, summary statistics are displayed: Participants: 1, Topics: 1, Posts: 1. Below this, a table titled 'All Topics' shows a single item: 'Topic Name' (topc1797: Task 1 Reports), 'Posts' (1), 'Views' (0), 'Created By' (Kevin O'Connor (FIRST)), and 'Created On' (10/04/2017 6:24 PM). The table has columns for Topic Name, Posts, Views, Created By, and Created On.

In the Task Reports section, you will see a Topic for each task. To see the reports and discussion for this task, click on the Topic Name. (Note: this example image only has a single Task).

Reporting Progress

The screenshot shows a forum topic titled 'Task 1 Reports'. The summary post, made by Kevin O'Connor (FIRST) on 10/04/2017 at 6:24 PM, contains the following text: 'Post reports for Task 1 here. Some examples of what you may want to include in your report: 1. How long did the software installation procedure take on your computer(s)? We would like you to time the installation of the software. 2. What problems or difficulties did you encounter? 3. Where were issues - What was the configuration of the computer(s) on which you installed the software? (manufacturer, model, processor type, processor speed, system memory, DVD model, operating system version) 4. What questions did you have during the process? 5. Any specific suggestions on improving the documentation? (Were any instructions unclear?) 6. Is there anything else you want to tell us related to this task?' At the bottom of the post, there are three buttons: 'Associate', 'Quote', and 'Reply'. A red arrow points to the 'Reply' button.

Each Topic will have a summary post at the top with suggested feedback. When you have completed the task, click **Reply** on this top summary post to post your report. Then compose or paste your report into the editor and click **Save**.

FRC 2019 Beta

! We strongly prefer reports be done as text posts as opposed to attachments.

Trackers - Reporting Bugs

For the 2019 Beta, the Trackers tab will be used only for filing bugs (different than the past few seasons). For reporting progress, see [Reporting Progress](#).

- ⚠** While we will certainly accept and respond to bugs in either location, for 2019 we would prefer bugs for WPILib C++/Java components (libraries, VSCode, Shuffleboard, etc.) be submitted to the appropriate [GitHub repositories](#).

Bug Tracker

Tracker Summary

Tracker Name	Summary
Bugs	0 Open, 0 Closed: 0 Total
Progress Reporting	0 Open, 0 Closed: 0 Total
Tasks	0 Open, 0 Closed: 0 Total
Code Coverage	0 Open, 0 Closed: 0 Total

Submit Artifact

Title:^{*} 1

Description:^{*} 2

Reported By:^{*} 3

Category:^{*} 4

Reported in Release: 5

Status

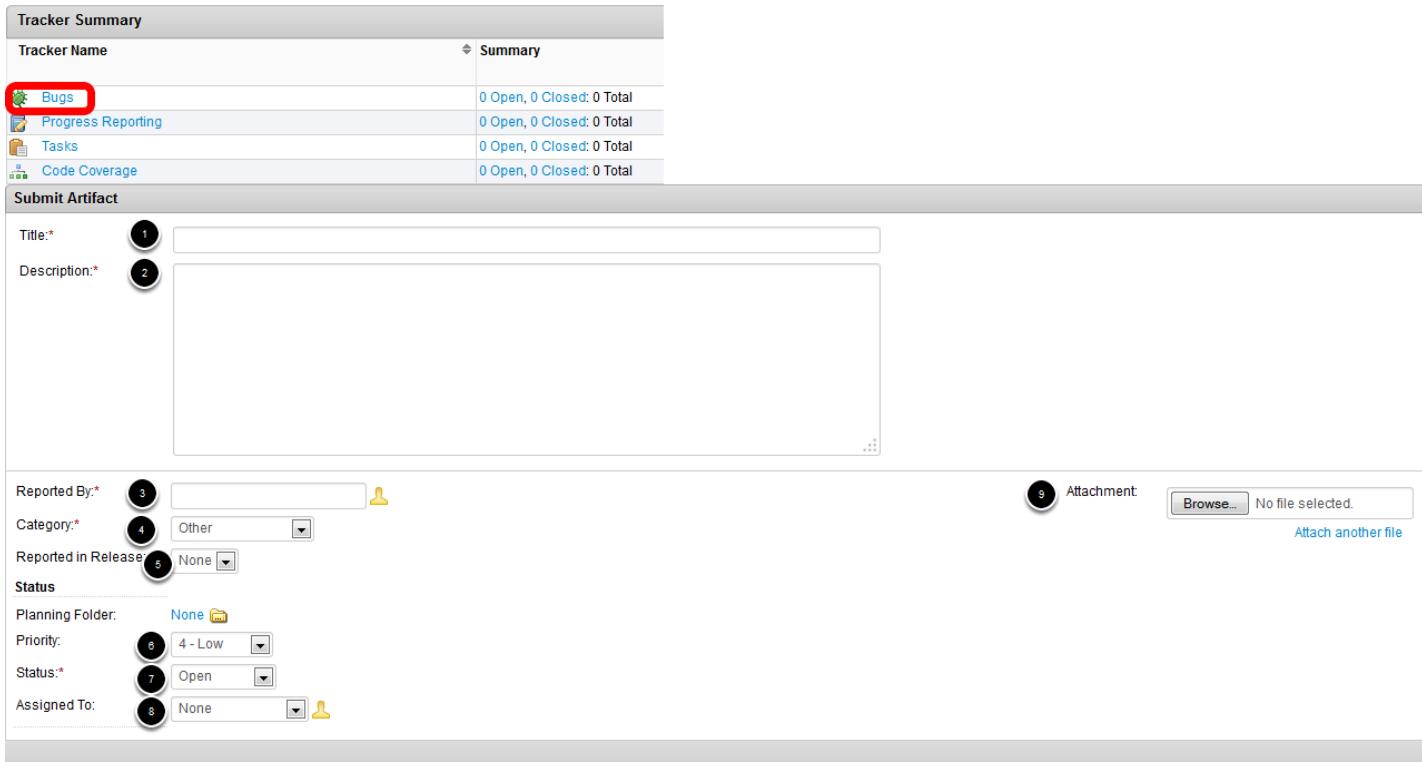
Planning Folder: 6

Priority: 7

Status:^{*} 8

Assigned To: 9

Attachment: No file selected. [Attach another file](#)



FRC 2019 Beta

The bug tracker is used for reporting bugs you discover in WPILib or other FRC software. If you are not sure if particular behavior is a bug or not, err on the side of reporting the bug (or at least posting in the Discussion section).

On the Trackers tab, click on Bugs to reach the Bug Tracker. To Submit a new bug, click **Submit New Artifact**. The fields of the artifact are:

1. Title - Give your bug a meaningful title which describes the issue
2. Description - Describe the bug including observed behavior, expected behavior and steps to reproduce
3. Reported By - Enter your TeamForge user name or click the silhouette to open the selector where you can select your username.
4. Category - Select the appropriate affected software from the dropdown list (LabVIEW WPILib, Java WPILib, Driver Station, etc.)
5. Reported in Release - Select which version of the file you were using, if available.
6. Priority - If desired you may estimate a priority based on bug severity. Leaving at the default priority does not mean the bug will not be addressed. All new bugs are analyzed by the Control System team and assigned an appropriate priority.
7. Status - When reporting a new bug this should be left as Open.
8. Assigned To - When reporting a bug this should be left blank. The Control System team uses this field to indicate who is working on a given artifact.
9. Attachment - Attach any additional file that may be appropriate, examples of appropriate files would be a patch for the bug, code to reproduce the bug (single file or zip) or an image of the bug (if appropriate/necessary).

Monitoring Via Email Notifications

You may wish to monitor specific items or even whole sections of the Beta project via email. TeamForge has a large degree of flexibility in setting up e-mail monitoring.

Monitoring Individual Items

The screenshot shows the 'Edit' screen for an artifact titled 'Team 0011 Java - Task 1'. The 'Description' field contains a detailed text about verifying software and firmware installation procedures. The 'Submitted By' field shows 'Charles T. Harris'. The 'Submitted On' and 'Last Modified' fields both show '09/12/2013 6:43 PM GMT'. The 'Status' dropdown is set to 'Open'. The 'Category' dropdown is set to 'All'. The 'Priority' dropdown is set to '3 - Medium'. The 'Assigned To' dropdown is set to 'None'. The 'Planning Folder' dropdown is set to 'Teams > 0011'. The 'Estimated Effort' and 'Remaining Effort' dropdowns are both set to '1'. The 'Attachments' section shows a comment text area and a file attachment area with a 'Browse...' button. The 'Comments' section shows a single comment from Charles T. Harris on 09/12/2013 at 6:43 PM GMT. At the bottom right, there are buttons for 'Cancel', 'Update And View', 'Update', 'Return', 'Users Is Monitoring', and 'Monitor'. A red arrow points to the 'Monitor' button.

You can choose to monitor a number of different individual items (forum threads, tracker artifacts, file releases, etc.) via e-mail. Items that can be subscribed to will have a Monitor button located at the bottom of the screen. Clicking this button will subscribe you to the item and notify you of changes via e-mail. To unsubscribe, click the Stop Monitoring button (which will replace the Monitor button when you are subscribed).

FRC 2019 Beta

Monitoring folders

The screenshot shows a 'Topics' list with various filters at the top. A red arrow points to a context menu that appears when clicking the 'Monitor' button. The menu includes options like 'Monitor Selected', 'Stop Monitoring Selected', 'Users Monitoring Selected', 'Monitor Current Folder' (which is highlighted with a red arrow), and 'Users Monitoring Folder'.

"Folders" (discussion sections, trackers, etc.) can also be monitored via e-mail. To monitor a folder click the dropdown arrow next to the Monitor button and select **Monitor Current Folder**.

Monitoring Project Sections

The screenshot shows the 'Monitoring' tab selected in the navigation bar. A red arrow points to the 'Monitored Applications' tab. On the left, a list of projects is shown, with 'FRC2019Beta' selected (circled with number 3). A red arrow points to this selection. Number 4 points to the 'Edit Monitoring Subscriptions and Preferences' link. Number 5 points to the 'Monitor Project' section where checkboxes for 'All Applications' and other sections like 'Trackers', 'Tasks', etc., are listed. Number 6 points to the 'Save' button at the bottom right.

It is also possible to set up for entire sections of the site. To set up this monitoring:

1. Click My Workspace
2. Click on the Monitoring Tab
3. Select the project from the list on the left
4. Click on the Monitored Applications tab
5. Check the box for any section you wish to subscribe to
6. Click Save

FRC 2019 Beta

Monitoring Settings

The screenshot shows the 'My Workspace' section of the TeamForge interface. At the top, there are navigation links: 'Project' (circled 1), 'My Workspace' (selected), 'Admin', 'History', and 'More'. Below this is a toolbar with 'My Page', 'Dashboard', 'Projects', 'Monitoring' (circled 2), and 'My Settings'. The main area displays 'User Details: Kevin O'Connor' with a placeholder profile picture. A note says 'No detail specified'. Below this are user details: User Name (koconnor), Full Name (Kevin O'Connor), Email Address (koconnor@usfirst.org), Alternate Emails, Status (Active), License Type (ALM), Organization (FIRST), Site Admin (Yes), Restricted User (No), Date Created (09/13/2013), Last Login (09/13/2013 7:29 PM), Date Time Format (MM/dd/yyyy h:mm a z), and Locale (English). At the bottom right are 'Change Password' and 'Edit' buttons. Below the user details is a 'User Preferences' tab bar with 'User Preferences' (selected), 'Authorization Keys', 'Roles', and 'User Group Membership'. Under 'User Preferences', there are sections for 'Notifications on Monitored Items' (circled 3) and 'Include My Own Updates in My Notifications'. The 'File Encoding for Export' dropdown is set to 'UTF-8'. At the bottom right are 'Cancel' and 'Save' buttons (circled 4).

You can set your overall notification settings to e-mail you per change or to send you a daily digest email with all changes. To do this:

1. Click My Workspace
2. Click My Settings
3. Select the desired setting
4. Click Save

Per Section Notification Settings

The screenshot shows the 'Edit Monitoring Subscriptions and Preferences' page. At the top, there are navigation links: 'Project' (circled 1), 'My Workspace' (selected), 'Admin', 'History', and 'More'. Below this is a toolbar with 'My Page', 'Dashboard', 'Projects', 'Monitoring' (circled 2), and 'My Settings'. The main area has tabs: 'Edit Monitoring Subscriptions and Preferences' (selected), 'All Monitored Objects', 'Monitored Applications', and 'Email Notification Preference' (circled 4). On the left, there's a sidebar with 'All Monitored Objects' and 'Projects: Mine' (circled 3). It lists 'Alalea', 'FRC2014Beta', 'Kevin's Test Project', 'SmartDashboard2', and 'WPILib'. In the center, there's a table for 'Email Notification Preference' (circled 5) with columns for Trackers, Tasks, Source Code, File Releases, Documents, Wiki, and Discussions, all set to 'Defined in My Settings'. At the bottom right are 'Cancel' and 'Save' buttons (circled 6).

You can also customize notifications on a per section basis. To do this:

FRC 2019 Beta

1. Click My Workspace
2. Click Monitoring
3. Click on the FRC2019Beta project on the left side
4. Select the Email Notification Preferences tab
5. Change the settings for any section of the project you wish to have override your base settings.
6. Click Save

2019 Beta Testing - Tasks

2019 Beta Task Overview

This document provides an overview of the Tasks assigned for the 2019 Control System Beta Test. Further details on each Task can be found in the subsequent pages of this section.

Task 0 - Background and Preparation

The purpose of Task 0 is to complete some housekeeping tasks and familiarize yourself with the tools and documentation for the Beta Test before beginning. This task consists of 1 Objective:

1. Beta Test Tools and Documentation Familiarization

Task 1 - Install Software (Targeted Completion 11/2/18)

The purpose of Task 1 is to complete installation of the 2019 Control System software. This task consists of 1 Objective

1. Install the necessary software for your assigned Beta Test language (LabVIEW or VSCode+ plugins, NI FRC Update Suite, CTRE installer if using CANTalon SRX)

Task 2 - Basic Benchtop Test (Targeted Completion 11/9/18)

The purpose of Task 2 is to get the base 2019 Control System software set up and running. This helps minimize complexity to ease debugging of any issues that do occur. In this task you will set up the development environment for your programming language, configure the roboRIO, load and debug a simple program. This Task consists of 2 Objectives:

1. Configure the roboRIO
2. Create, Build and Load Default Program

Task 3 - Port Previous Robot Code (Targeted Completion 11/16/18)

The purpose of Task 3 is to port a previous year's robot code (2018 preferred) with the 2019 Control System software and bring it to full functionality. This Task consists of 2 Objectives and 1 Optional Objective:

1. Port Existing Robot Code to 2019 System
2. Wireless Operation of FRC Robot
3. (Optional) Offseason Competition with Existing FRC Robot

Task 4 - New Code and Advanced Feature Testing (Targeted Completion 11/30/18)

Note: Objectives in Task 4 may be completed in any order.

The purpose of Task 4 is to write new code in order to test features and functionality of the new Control System software. While we have written up a number of ideas, please feel free to come up with your own and report on what you tried/are trying and how it went.

1. Pathweaver (Motion profile generator UI) testing
2. Shuffleboard "builder" (create layout from Robot code) testing
3. Camera streaming robustness, especially in LabVIEW (multiple cameras, disconnect/reconnect, etc.)

International Team Note

A number of our international teams may have access to computers with foreign language Operating Systems installed (Spanish, French, Hebrew, etc.) This note describes some additional feedback that would be helpful to us if you have access to such a machine to use as part of your Beta test effort.

Tasks

If you have access to a computer with a foreign language version of Windows, it would be helpful to our Beta Test effort to utilize that machine as part of your Beta testing. Specifically, try testing each of the unique software aspects (installation, imaging the roboRIO, running the DS, creating projects, building and downloading code, debugging code, programming a radio, etc.).

Feedback

In your task reports for the appropriate tasks please indicate that you completed the task with a machine with a foreign language OS.

Task 0: Beta Test Tools and Documentation Familiarization

Task: Learn about the tools and documentation sources for the 2019 FRC Beta

Beta Test Tools

The 2019 FRC Control System Beta Test will primarily be administered through a TeamForge project. If you have not been part of an Alpha or Beta test effort in the last few years, please take some time to read through the following articles to learn more about the tool being used:

[Accessing the 2019 Beta Project](#)

[Trackers - Reporting Bugs](#)

[Monitoring Via Email Notifications](#)

⚠ Note: For 2019 we are asking that issues for GitHub developed components (basically everything other than the DS/Image/LabVIEW libraries) be submitted on [GitHub Issues for the appropriate repository](#)

Beta Test Documentation

There will be 2 primary sources of Beta Test specific documentation. The first source is this ScreenSteps manual which will primarily contain things like task instructions. The Beta Test ScreenSteps page will be publicly accessible without a username or password.

The other source of documentation will be the Documents section of the TeamForge project. For more information on how to access that, see [Accessing the 2019 Beta Project](#).

Please read through the documentation in this ScreenSteps manual to understand the tasks associated with the Beta Test.

Desired Feedback

No report is necessary for this task.

Task 1 - Objective 1: 2019 Software Installation

Task: Choose, acquire, and install the software required for developing FRC software in the language you have been assigned (C++, Java or LabVIEW).

Overview

The 2019 FRC Control System can be programmed in LabVIEW, Java or C/C++. Teams should use the language they have been assigned for the Beta test.

Note: For this task, there is supplemental documentation provided in TeamForge for all languages. See the Documents and File Releases sections on TeamForge.

Desired Feedback

This Objective is part of Task 1. Please keep the following questions in mind as you complete the Objective and include this information, as appropriate, in your Task 1 report.

1. How long did the software installation procedure take on your computer(s)? We would like you to time the installation of the software.
2. What problems or difficulties did you encounter?
3. If there were issues - What was the configuration of the computer(s) on which you installed the software? (manufacturer, model, processor type, processor speed, system memory, DVD model, operating system version)
4. What questions did you have during the process?
5. Any specific suggestions on improving the documentation? (Were any instructions unclear?)
6. Is there anything else you want to tell us related to this task?

Task 2 - Objective 1: Configure the roboRio

Task: Image and configure your roboRIO controller.

Overview

Before using the roboRIO controller with 2019 software, it must be imaged with the 2019 image and configured with your FRC team number. As part of the purpose of this Beta test is to test the documentation and its ease of use, no direct links to setup instructions will be provided. Instructions for configuring the roboRIO are available on the [Getting Started with the 2018 Control System](#) page.

Note, the image version in the documentation may not be updated yet. You should use the latest 2019 image available after installing the latest 2019 Beta Update.

Desired Feedback

This Objective is part of Task 2. Please keep the following questions in mind as you complete the Objective and include this information, as appropriate, in your Task 2 report.

1. What problems or difficulties did you encounter?
2. What questions did you have during the process?
3. Any specific suggestions on improving the documentation? (Were any instructions unclear?)
4. Is there anything else you want to tell us related to this task?

Task 2 - Objective 2: Create, Build, and Load Program for Benchtop Testing

Task: To create, build, load and run a basic robot program (template of your choosing) on the roboRIO benchtop setup.

Overview

The purpose of this task is to verify the ability to create and download a basic robot program from the chosen template in the assigned programming language. This will verify that the development environment are set up correctly and working properly before moving on to a more complex system. For this task you should create, build and download a basic program to the roboRIO to run at startup, then verify that you are able to communicate with and enable the program using the Driver Station.

As part of the purpose of this Beta test is to test the documentation and its ease of use, no direct links to setup instructions will be provided. Instructions for all languages are available on the [Getting Started with the 2018 Control System](#) page.

Desired Feedback

This Objective is part of Task 2. Please keep the following questions in mind as you complete the Objective and include this information, as appropriate, in your Task 2 report.

1. What problems or difficulties did you encounter?
2. What questions did you have during the process?
3. Any specific suggestions on improving the documentation? (Were any instructions unclear?)
4. Is there anything else you want to tell us related to this task?

Task 3 - Objective 1: Port Existing Robot Code to 2019 System

Task: Port 2018 Robot code to 2019 system.

Overview

The purpose of this task is to port your 2018 robot code to the 2019 development environment and bring it to a fully functional state. Make sure to post in the discussions or file a bug in the tracker for any items that do not seem to be working properly.

Desired Feedback

This Objective is part of Task 3. Please keep the following questions in mind as you complete the Objective and include this information, as appropriate, in your Task 3 report.

1. Note any required changes to your robot program not detailed in the documentation for porting robot code.
2. Using the Charts tab of the Driver Station, report the resource usage of your robot code (roboRIO CPU usage, free RAM)
3. What problems or difficulties did you encounter?
4. What questions did you have during the process?
5. Any specific suggestions on improving the documentation? (Were any instructions unclear?)
6. Is there anything else you want to tell us related to this task?

(Optional) Task 3 - Objective 3: Offseason Competition with Existing FRC Robot

This Objective is optional and does not need to be completed in order to continue with the other Tasks and Objectives.

Task: Participate in an offseason event with a 2017 robot which has been retrofitted with the 2018 FRC Control System software.

Overview

The purpose of this task is to test the 2018 Control System in a competition environment.

Desired Feedback

This Objective is part of Task 3. Please keep the following questions in mind as you complete the Objective and include this information, as appropriate, in your Task 3 report.

1. What problems or difficulties did you encounter?
2. Is there anything else you want to tell us related to this task?

Task 4 - Objective 1: Testing New Features

Task: Test any new features or heavily modified code

Overview

The purpose of this task is to test any newly developed or heavily modified features for the 2019 season. This will help the Control System team find any bugs or issues with the new code. Below is a list of some of the new or modified features:

LabVIEW

- Camera Streaming robustness (camera connect/disconnect, network connect/disconnect)

C++\Java

- Pathweaver (motion profile generator UI)
- Shuffleboard "builder" robot code (class to build shuffleboard layout from robot code)

Desired Feedback

This Objective is part of Task 4. Please keep the following questions in mind as you complete the Objective and include this information, as appropriate, in your Task 4 report.

1. Describe what features you tested and how you tested them.
2. Describe any bugs you encountered.
3. What problems or difficulties did you encounter?
4. What questions did you have during the process?
5. Any specific suggestions on improving the documentation? (Were any instructions unclear?)
6. Is there anything else you want to tell us related to this task?

2019 Beta Specific Documentation (Beta Teams Only)

Installing VS Code for FRC

Windows

Offline Installer

 We are currently investigating 2 known issues:

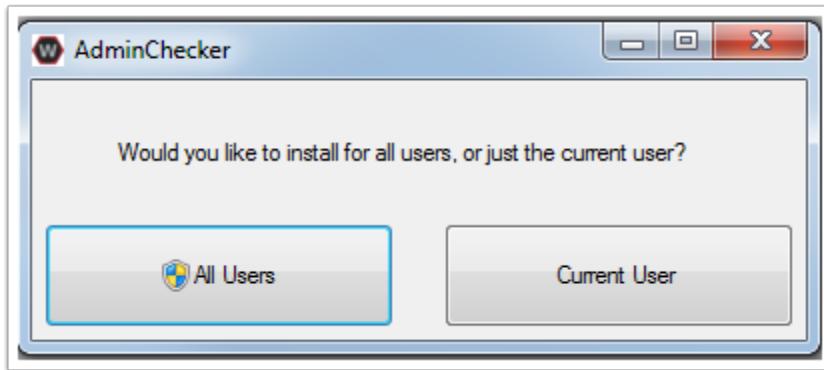
- The installer may not launch on Windows 7. Install the NI Update Suite first to get the necessary .NET dependency. We are hoping to resolve this to allow development only computers.
- The installer may throw a .NET error near the end of the install. This appears to prevent the Desktop icon from being installed and possibly some cleanup but the rest of the installation appears to complete.

Download the appropriate offline installer for your Windows installation (32 bit or 64 bit). If you're not sure, open the Control Panel -> System to check.

For Beta, these installers are found in the File Releases section of the Teamforge Beta project.

Double click on the installer to run it. If you see any Security warnings, click Run (Windows 7) or More Info->Run Anyway (Windows 8+)

Installation Type

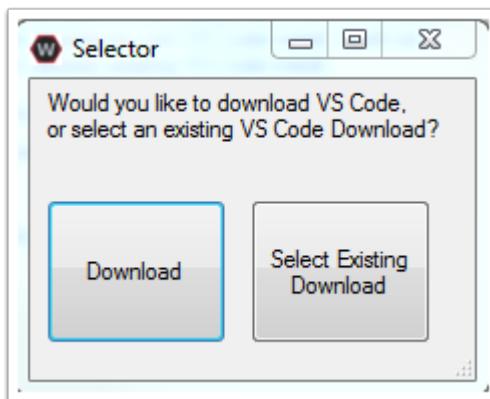


Choose whether to install for All Users on the machine or the Current User. The All Users option requires Admin privileges, but installs in a way that is accessible to all user accounts, the Current User install is only accessible from the account it is installed from.

If you select All Users, you will need to accept the security prompt that appears.

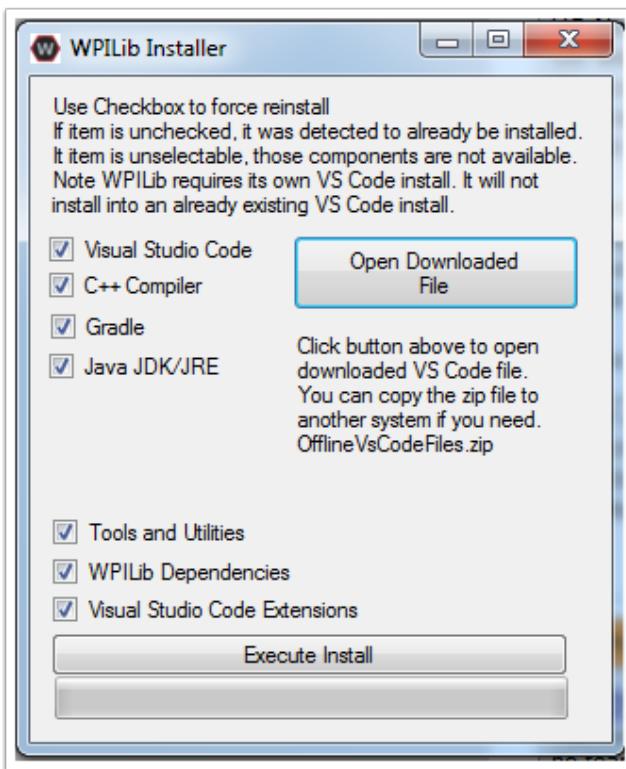
Download VSCode

For licensing reasons, the installer cannot contain the VSCode installer bundled in. Click Select/Download VSCode to either Download the VSCode installer or select a pre-downloaded copy. If you intend to install on other machines without internet connections, after the download completes, you can click Open Downloaded File to be taken to the zip file on the file system to copy along with the Offline Installer.



Execute Install

Make sure all checkboxes are checked (unless you have already installed 2019 WPILib software on this machine and the software unchecked them automatically), then click Execute Install.



What's Installed?

The Offline Installer installs the following components:

- Visual Studio Code - The supported IDE for 2019 robot code development. The offline installer sets up a separate copy of VSCode for WPILib development, even if you already have VSCode on your machine. This is done because some of the settings that make the WPILib setup work may break existing workflows if you use VSCode for other projects.
- C++ Compiler - The toolchains for building C++ code for the roboRIO
- Gradle - The specific version of Gradle used for building/deploying C++ or Java robot code

FRC 2019 Beta

- Java JDK/JRE - A specific version of the Java JDK/JRE that is used to build Java robot code and to run any of the Java based Tools (Dashboards, etc.). This exists side by side with any existing JDK installs and does not overwrite the JAVA_HOME variable
- WPILib Tools - SmartDashboard, Shuffleboard, Robot Builder, Outline Viewer, Pathweaver
- WPILib Dependencies - OpenCV, etc.
- VSCode Extensions - Extensions for robot code development in VSCode

What's Installed - Continued

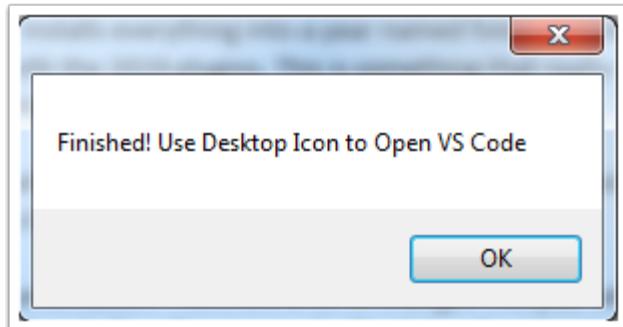
The Offline Installer also installs a Desktop Shortcut to the WPILib copy of VSCode and sets up a command shortcut so this copy of VSCode can be opened from the command line using the command "frccode2019"

Both of these reference the specific year as the WPILib C++\Java tools will now support side-by-side installs of multiple environments from different seasons.



Finished!

When the installer completes, you will now be able to open and use the WPILib version of VSCode. If you are using any 3rd party libraries, you will still need to install those separately before using them in robot code.



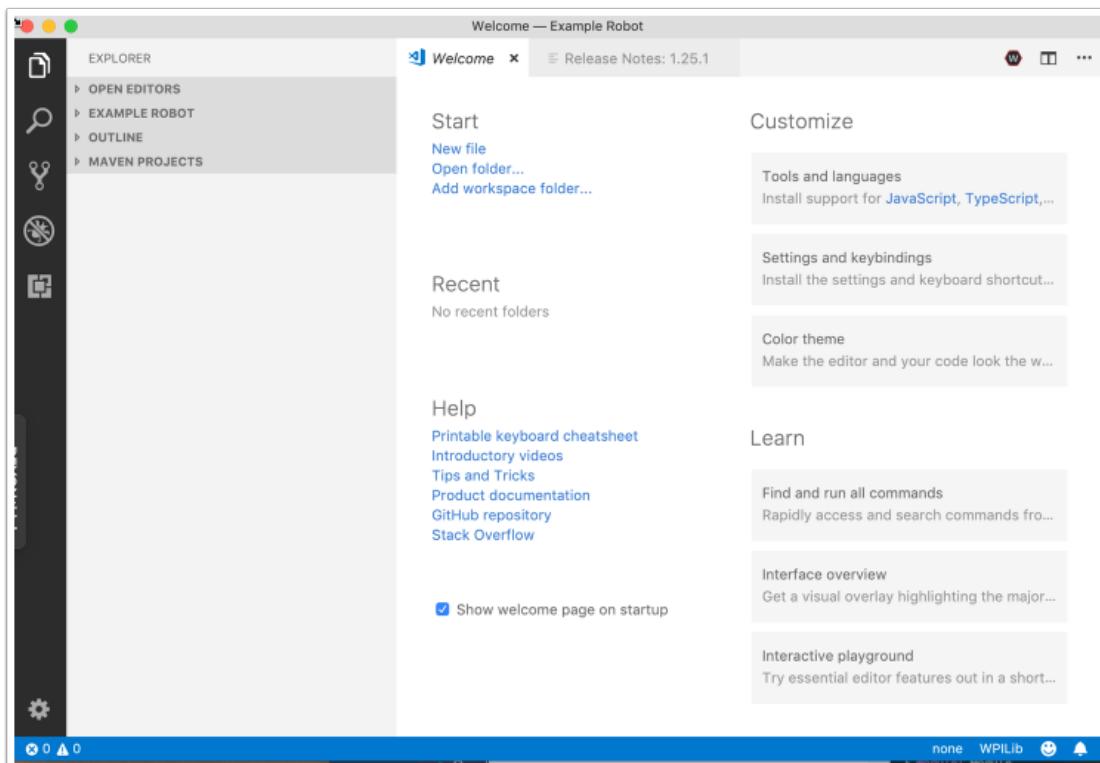
VS Code Basics and WPILib in VS Code

Microsoft's Visual Studio Code (VS Code) is the new supported IDE for C++ and Java development in FRC, replacing the Eclipse IDE used from 2015-2018. This article introduces some of the basics of using VS Code and the WPILib extension.

⚠ Note: If you used the publicly available VSCode Alpha, you should create a new project, re-import your Eclipse project with the Beta. There were breaking changes made to some of the configuration files (such as build.gradle) between releases.

The rest of this article is currently still the same as Alpha.

VS Code Welcome Page



FRC 2019 Beta

When VS Code first opens, you are presented with a Welcome page. On this page you will find some quick links that allow you to customize VS Code as well as a number of links to help documents and videos that may help you learn about the basics of VS Code as well as some tips and tricks.

You may also notice a small WPILib logo way up in the top right corner. This is one way to access the features provided by the WPILib extension (discussed further below).

VS Code User Interface

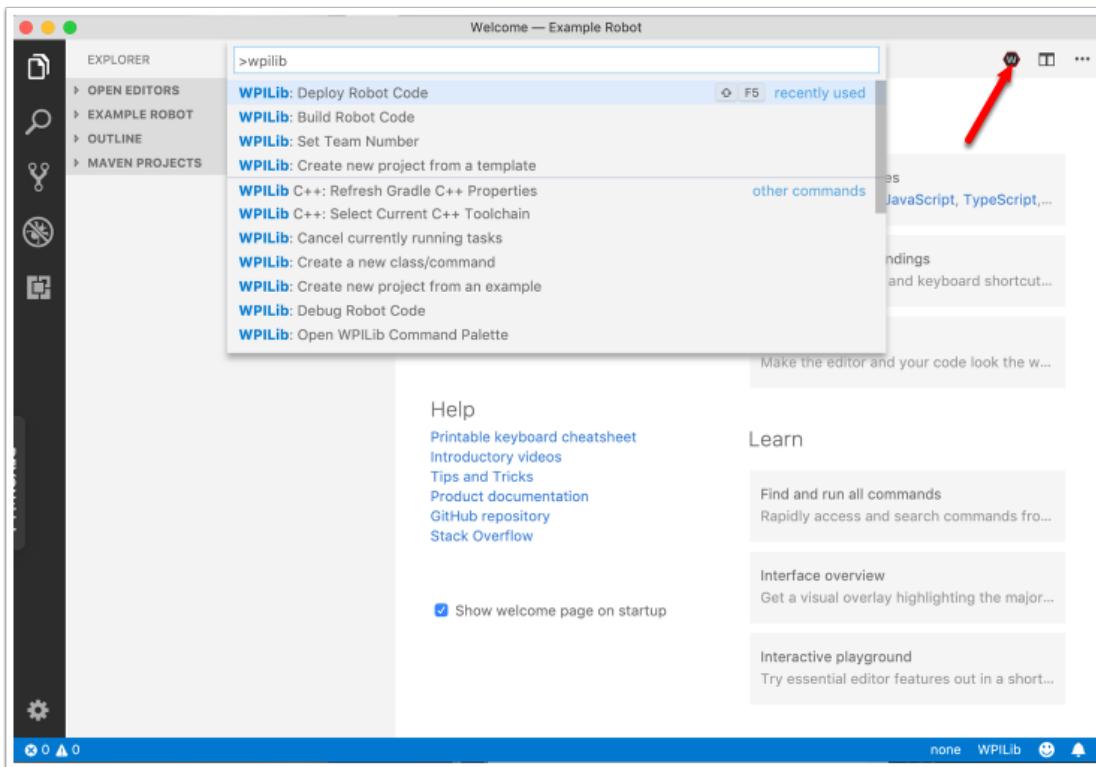
The most important link to take a look at is probably the basic [User Interface document](#). This document describes a lot of the basics of using the VS Code UI and provides the majority of the information you should need to get started using VS Code for FRC.

VS Code Command Palette

The Command Palette can be used to access or run almost any function or feature in VS Code (including those from the WPILib extension). The Command Palette can be accessed from the View menu or by pressing Ctrl+Shift+P (Cmd+Shift+P on Mac). Typing text into the window will dynamically narrow the search to relevant commands and show them in the dropdown.

In the following example "wpilib" is typed into the search box after activating the Command Palette, and it narrows the list to functions containing WPILib.

WPILib Extension



The WPILib extension provides the FRC specific functionality related to creating projects and project components, building and downloading code to the roboRIO and more. You can access the WPILib commands one of two ways:

- By typing "WPILib" into the Command Palette
- By clicking on the WPILib icon in the top right of most windows. This will open the Command Palette with "WPILib" pre-entered

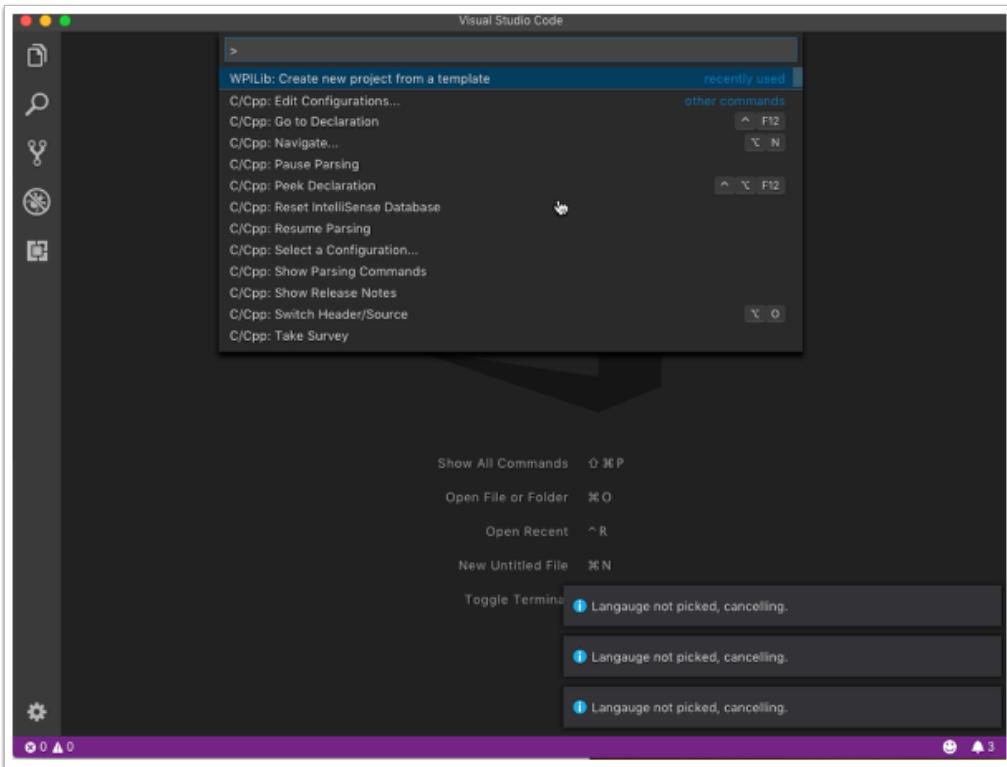
For more information about specific WPILib extension commands, see the other articles in this chapter.

Creating a new WPILib project in VS Code

In this article we will create a new WPILib project in VS Code. In this example we will be making a Command Based Robot, however the same methods apply to creating a project from any of the existing templates or examples.

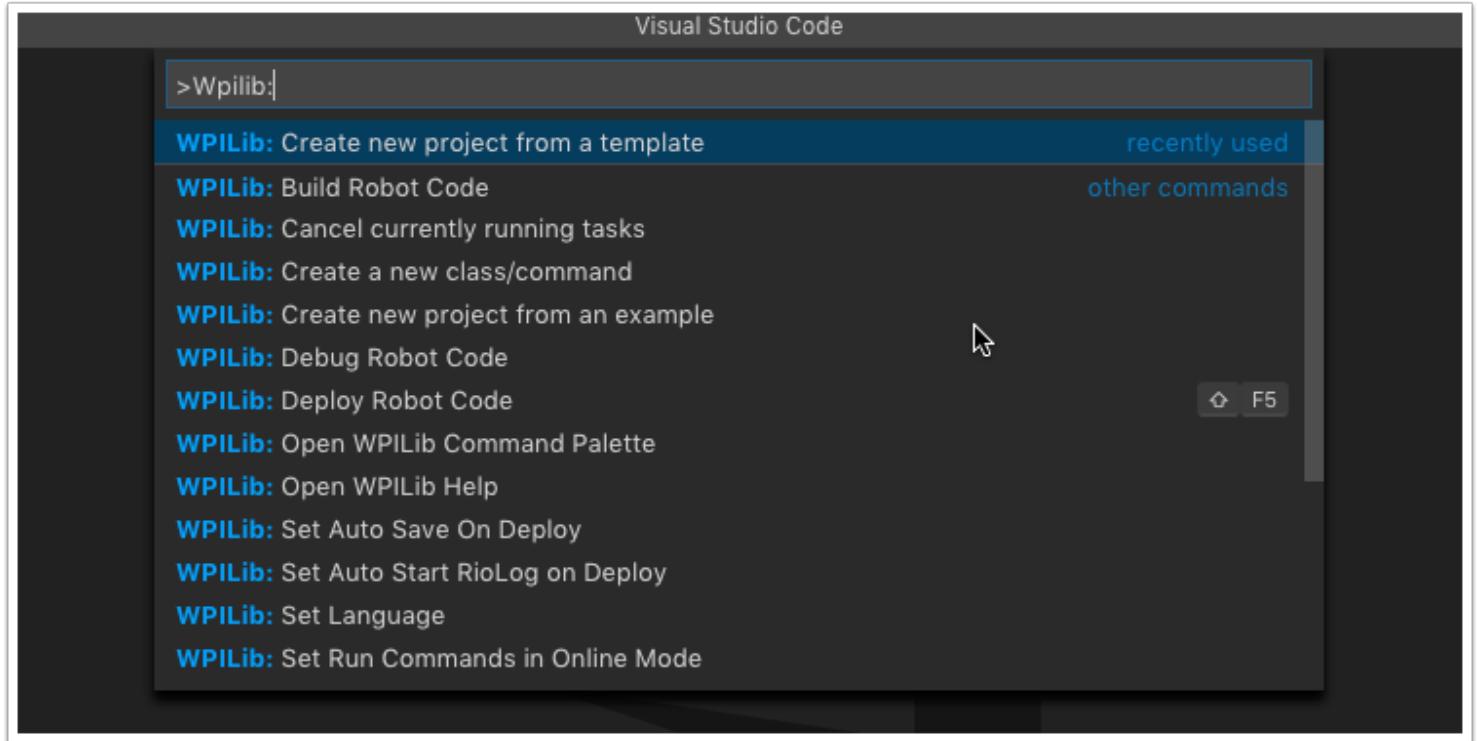
Accessing The Command Palette

Clicking Ctrl+Shift+P will open the command palette. The command palette contains the WPILib commands for creating and interacting with projects.



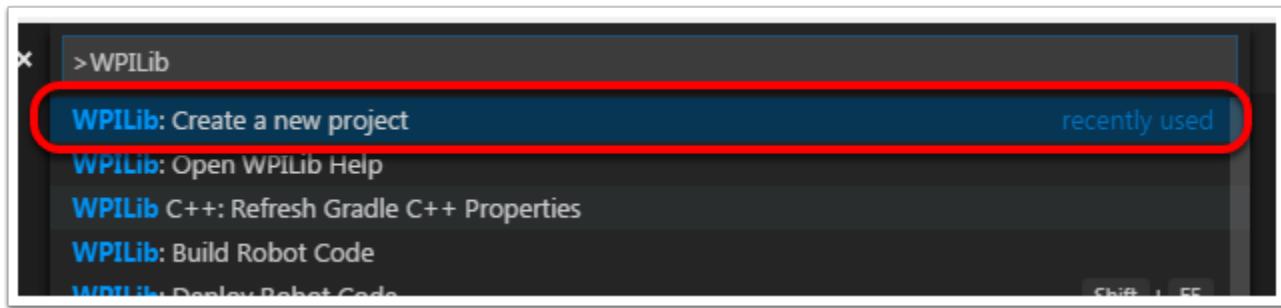
Accessing The WPILib Commands

All WPILib commands start with "WPILib:", so in order to access the WPILib commands type "WPILib:" into the command palette search bar.

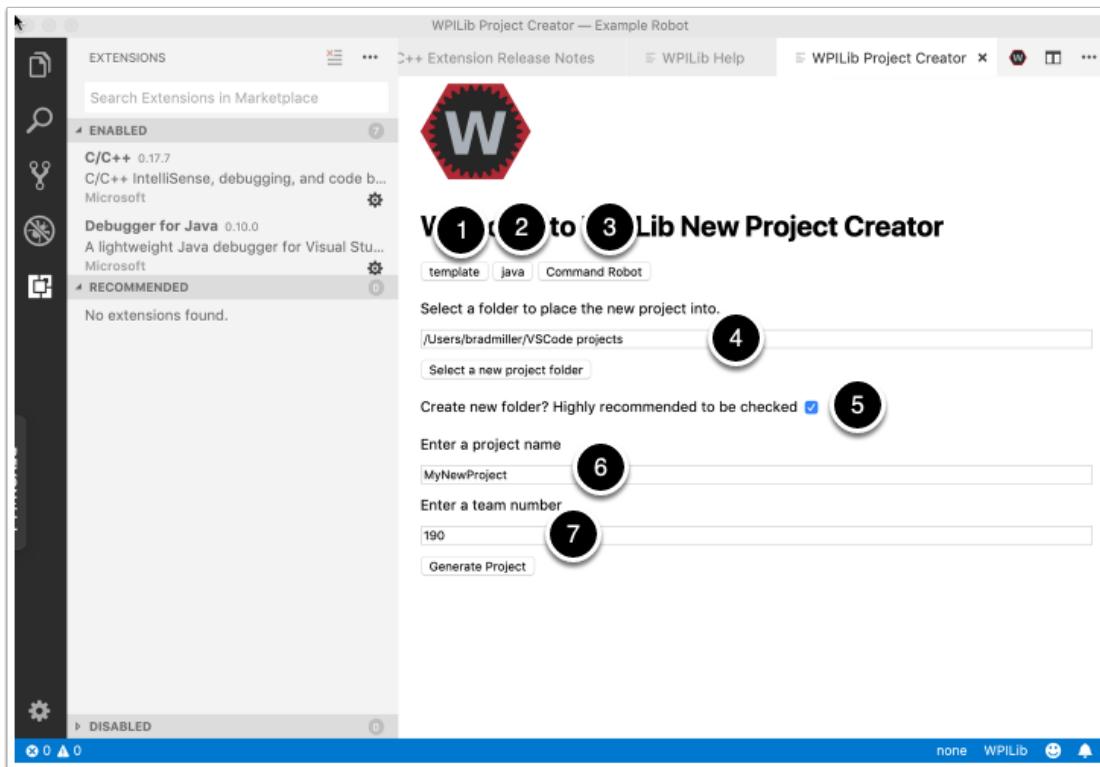


Creating A New WPILib Project

In order to create a new project select the "Create a new project" command. This will show a form with a number of fields where you enter the information required to create your new project.



New project creator window



The steps to create the new project are outlined here:

1. Select the kind of project you want to create. It can be an example project or one of the template projects provided by WPIlib.
2. Select the language that you are using for your project.
3. In the case of a template - select the template type (Timed robot, Iterative robot, Command robot, etc.)
4. Select the folder to place the project.
5. If the "Create new folder" checkbox is checked, a new folder named with the project name is created in the supplied folder. If the checkbox is NOT checked, then the folder supplied is assumed to be empty (will give an error if not) and the project files will be placed into that directory.
6. The project name is used in the project and also to optionally create the folder to place it if the checkbox from the previous step is checked.
7. The team number for the project. This will be used for package names and to locate your robot when deploying code.

FRC 2019 Beta

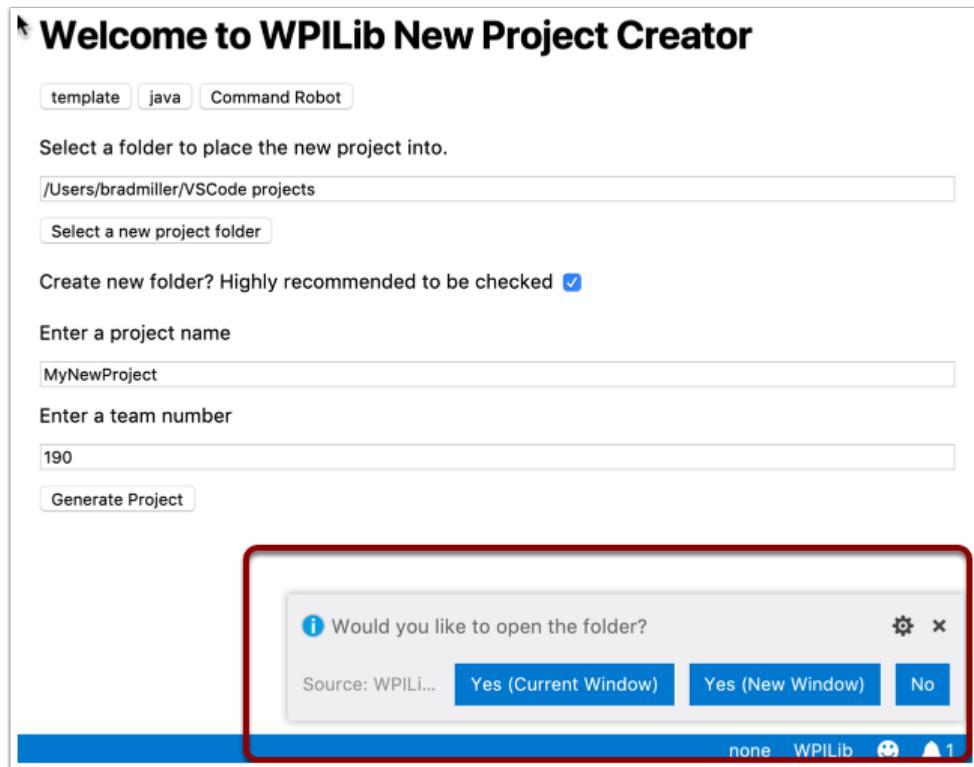
And last, click "Generate Project" and VS Code will create the project in the location specified.

Click the "Select a project type" button and then select your desired type from the dropdown.

- i** Note: If there are any errors generating the project (such as trying to use a non-empty folder with the checkbox unchecked), they will pop up in the bottom right corner of the screen.

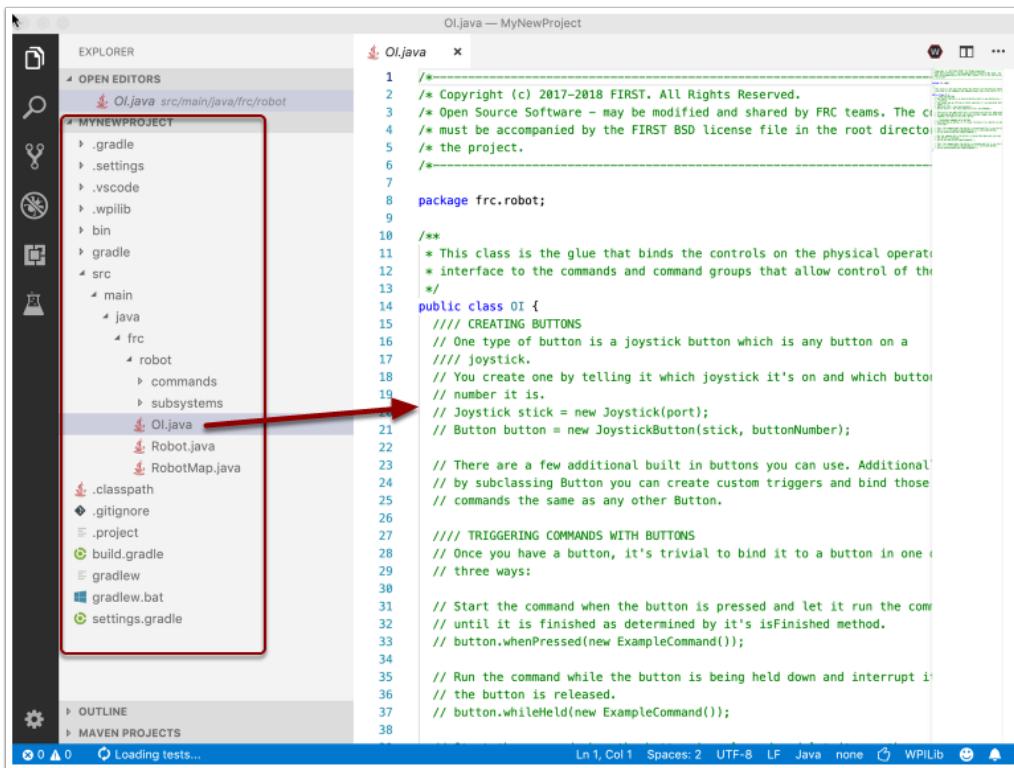
Opening The New Project

After successfully creating your project, VS Code will give you the option of opening the project as shown below. You can choose to do that now or later by typing Ctrl-O (Command+O on mac) and select the folder where you saved your project.

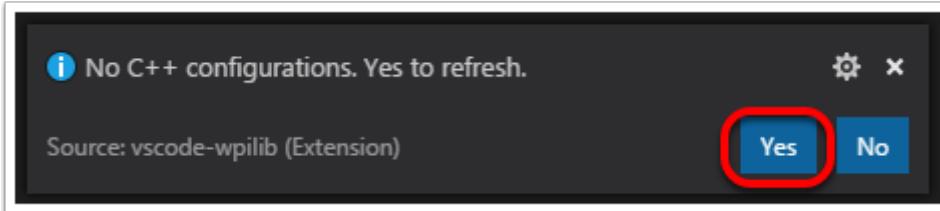


FRC 2019 Beta

Once opened you will see the project hierarchy on the left. Double clicking on the file will open that file in the editor.



C++ Configurations (C++ Only)



For C++ projects, there is one more step to set up IntelliSense. Whenever you open a project, you should get a pop-up in the bottom right corner asking to refresh C++ configurations, click Yes to setup IntelliSense.

Building and Deploying Robot Code

To build the robot project, do one of:

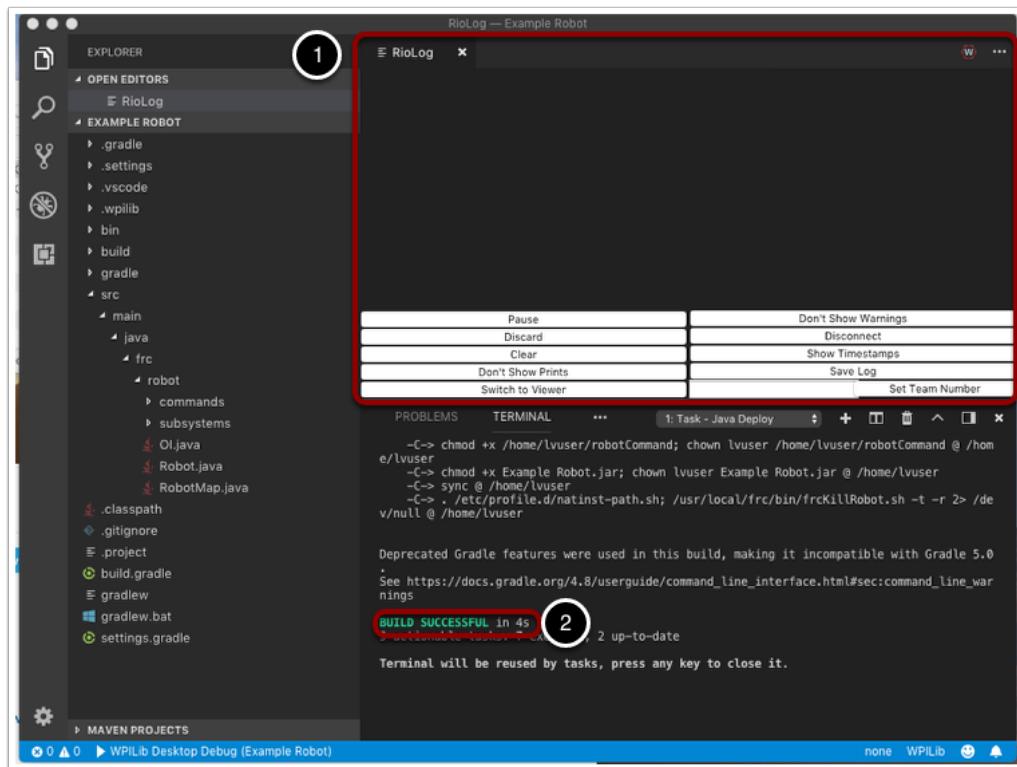
1. Open the Command Palette and select "WPILib: Build Robot Code"
2. Open the shortcut menu indicated by the ellipses in the top right corner of the VS Code window and select "Build Robot Code"
3. Right-click on the build.gradle file in the project hierarchy and select "Build Robot Code"

The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows the project structure for "MyNewProject". The "build.gradle" file is selected and highlighted with a red box.
- EDITOR:** Displays the contents of the "build.gradle" file. The code includes definitions for Java plugins, a robot class, targets for RoboRIO, and deployment logic.
- STATUS BAR:** Shows the message "BUILD SUCCESSFUL in 6s" and "2 actionable tasks: 2 executed".
- TOP BAR:** Includes icons for file operations and a three-dot menu.

Deploy robot code by selecting "Deploy Robot Code" from any of the three locations from the previous instructions. That will build (if necessary) and deploy the robot program to the roboRIO. If successful, you see a "Build Successful" message (2) and the RioLog will open with the console output from the robot program as it runs.

FRC 2019 Beta

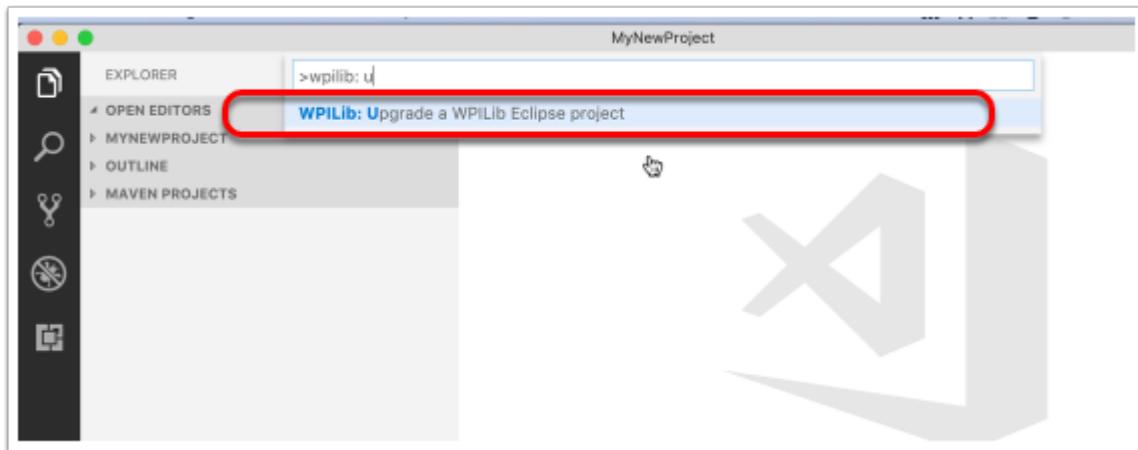


Importing an Eclipse project into VS Code

To make it easy for teams to use existing projects with the new IDE, we have implemented a wizard for importing Eclipse projects into VS Code. This will generate the necessary Gradle components and load the project into VS Code.

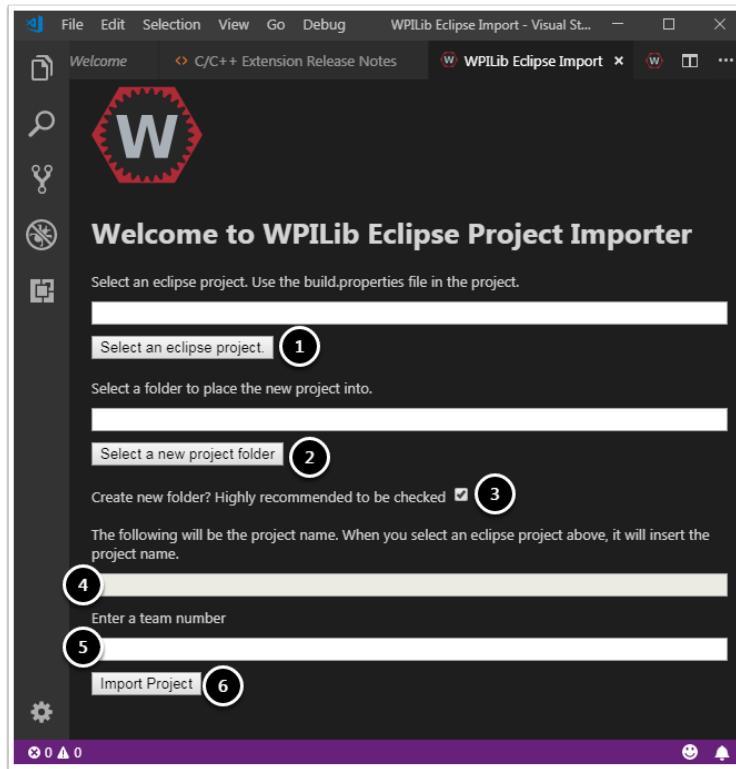
Launch the Import Wizard

Press Ctrl+Shift+P and type "WPILib" or click the WPILib icon to locate the WPILib commands. Select "Upgrade a WPILib Eclipse Project.



You'll be presented with the WPILib Eclipse Project Upgrade window. This is similar to the process of creating a new project and the window and the steps are shown below.

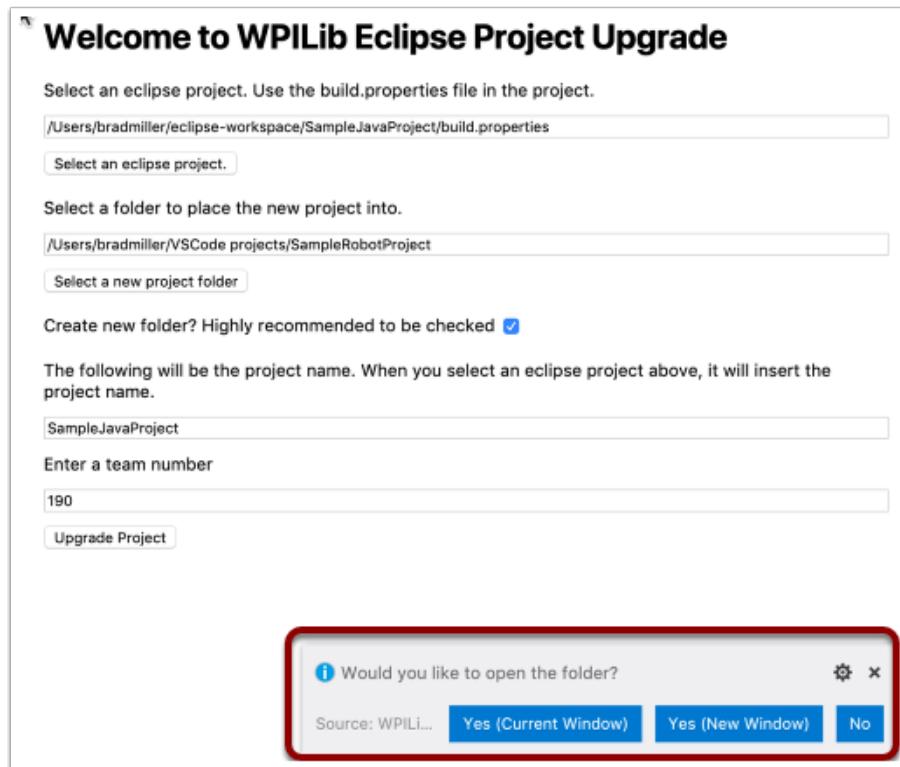
FRC 2019 Beta



Perform the following steps to fill in the Eclipse Project Upgrade window:

1. Select the eclipse project to convert. Select the build.properties file in the root directory of the eclipse project.
2. Fill in the new project folder by pressing the "Select a new project folder" button.
3. If the "Create new folder" checkbox is checked, then the project will be stored in a new folder under the one selected in 2. If it is not checked, then the project will be placed in the folder specified. It must be empty in that case.
4. Enter the name of the new project.
5. Enter the team number for the creation of the project and for the robot deployment.
6. And finally, click "Upgrade Project" to begin the upgrade.

The eclipse project will be upgraded and copied into the new project directory from step 3 above. You can then either open the new project immediately or open it later using the Ctrl-O (or Command-O for Mac) shortcut.



C++ Configurations (C++ Only)

For C++ projects, there is one more step to set up IntelliSense. Whenever you open a project, you should get a pop-up in the bottom right corner asking to refresh C++ configurations, click Yes to setup IntelliSense.

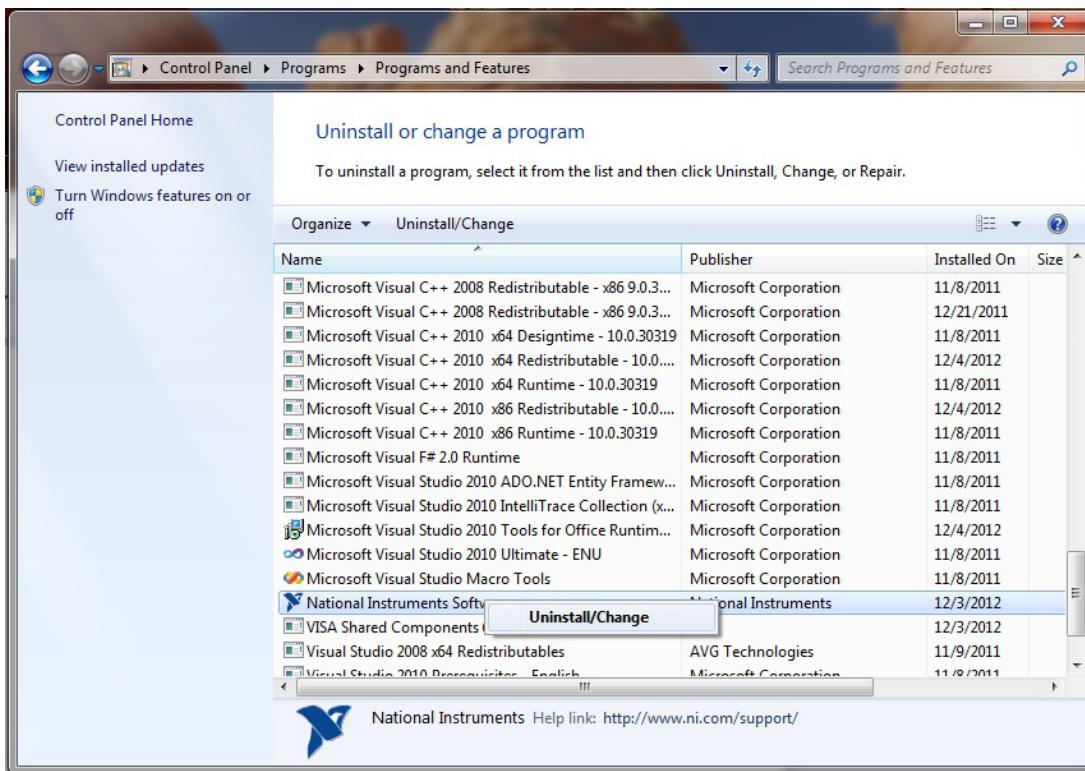
Installing LabVIEW for FRC 2019 (LabVIEW only)



Note: This installation is for teams programming in LabVIEW or using NI Vision Assistant only. C++ and Java teams not using these features do not need to install from the DVD.

Download and installation times will vary widely with computer and internet connection specifications, however note that this process involves a large file download and installation and will likely take at least an hour to complete.

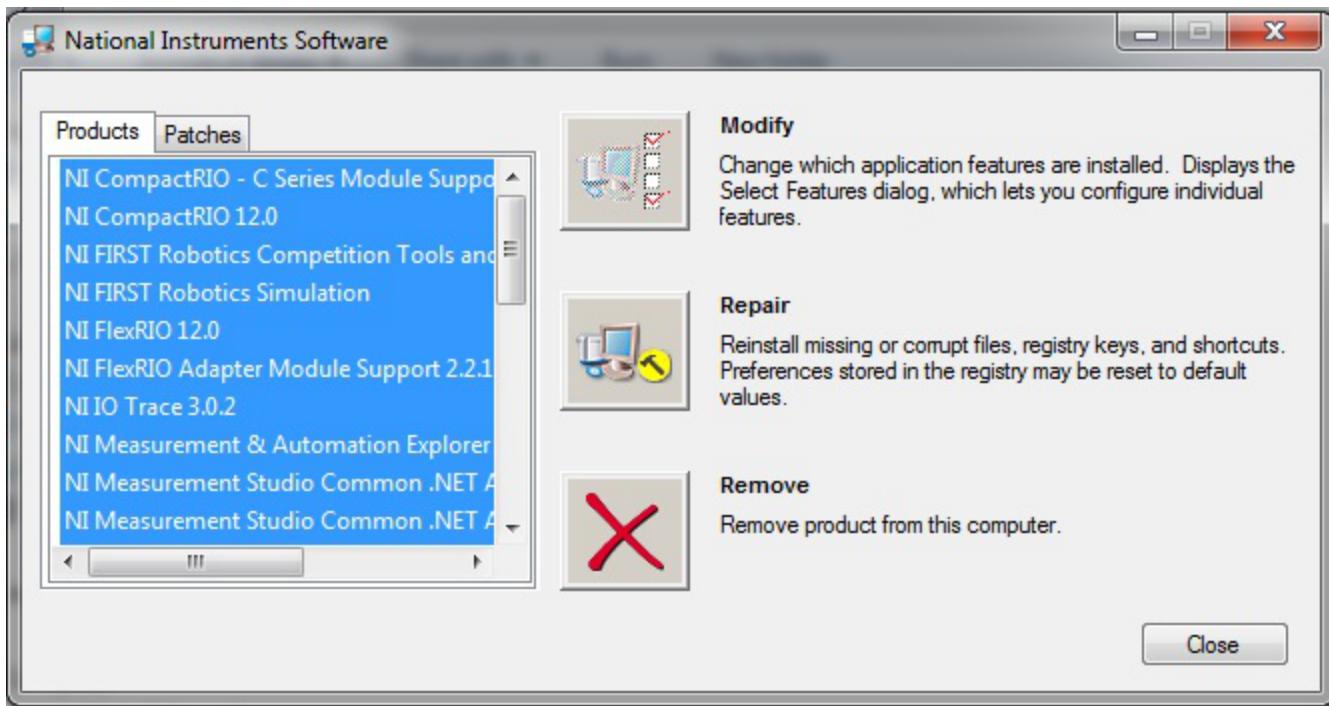
Uninstall Old Versions (Recommended)



NOTE: If you wish to keep programming cRIOs you will need to maintain an install of LabVIEW for FRC 2014. The LabVIEW for FRC 2014 license has been extended. While these versions should be able to co-exist on a single computer, this is not a configuration that has been extensively tested.

Before installing the new version of LabVIEW it is recommended to remove any old versions. The new version will likely co-exist with the old version, but all testing has been done with FRC 2019 only. Make sure to back up any team code located in the "User\LabVIEW Data" directory before un-installing. Then click Start >> Control Panel >> Uninstall a Program. Locate the entry labeled "National Instruments Software", right-click on it and select Uninstall/Change.

Select Components to Uninstall



In the left pane of the dialog box that appears, select all entries. The easiest way to do this is to click the top entry to highlight it, then scroll down to the bottom entry, press and hold shift and click on the last entry then release shift. Click Remove. Wait for the uninstaller to complete and reboot if prompted.

Getting LabVIEW Installer

Either locate and insert the LabVIEW USB Drive or download the LabVIEW 2019 installer from [the File Releases section of the Teamforge Beta project](#).

If downloaded, right click on the downloaded file ([FRC_Merged_DVD.zip](#)) and select Extract All.

Note: This is a large download (~5GB). It is recommended to use a fast internet connection and to use the NI Downloader to allow the download to resume if interrupted.

Installing LabVIEW

National Instruments LabVIEW requires a license. Each season's license is active until January 31st of the following year (e.g. the license for the 2019 season expires on January 31, 2020)

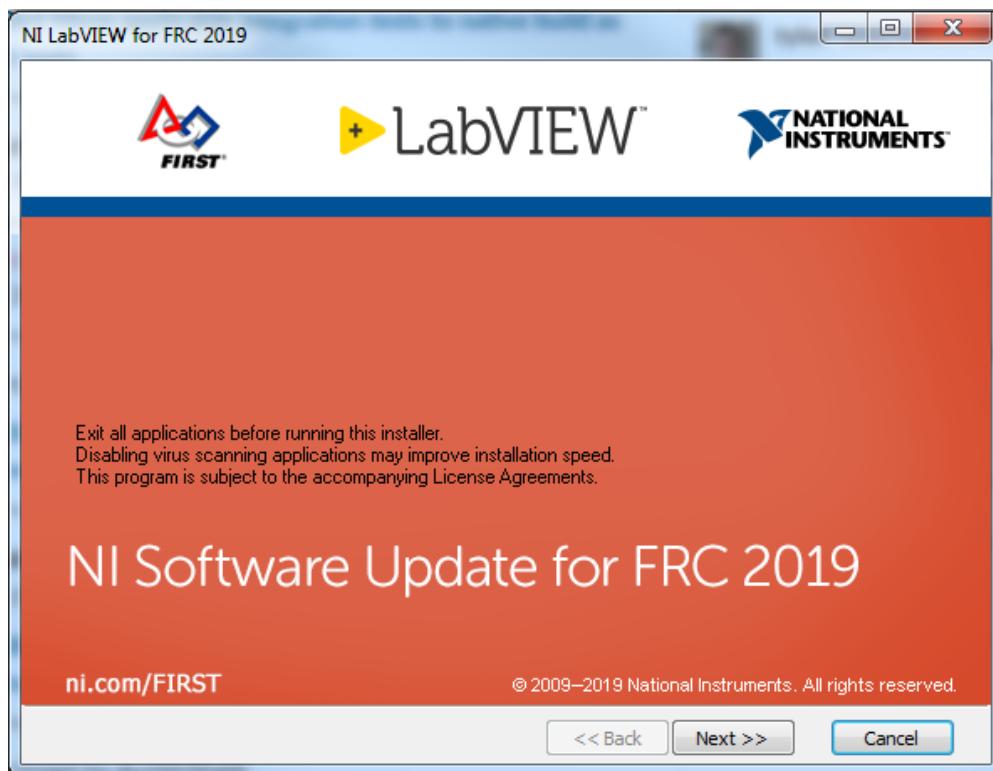
Teams are permitted to install the software on as many team computers as needed, subject to the restrictions and license terms that accompany the applicable software, and provided that only team members or mentors use the software, and solely for FRC. Rights to use LabVIEW are governed solely by the terms of the license agreements that are shown during the installation of the applicable software.

Welcome



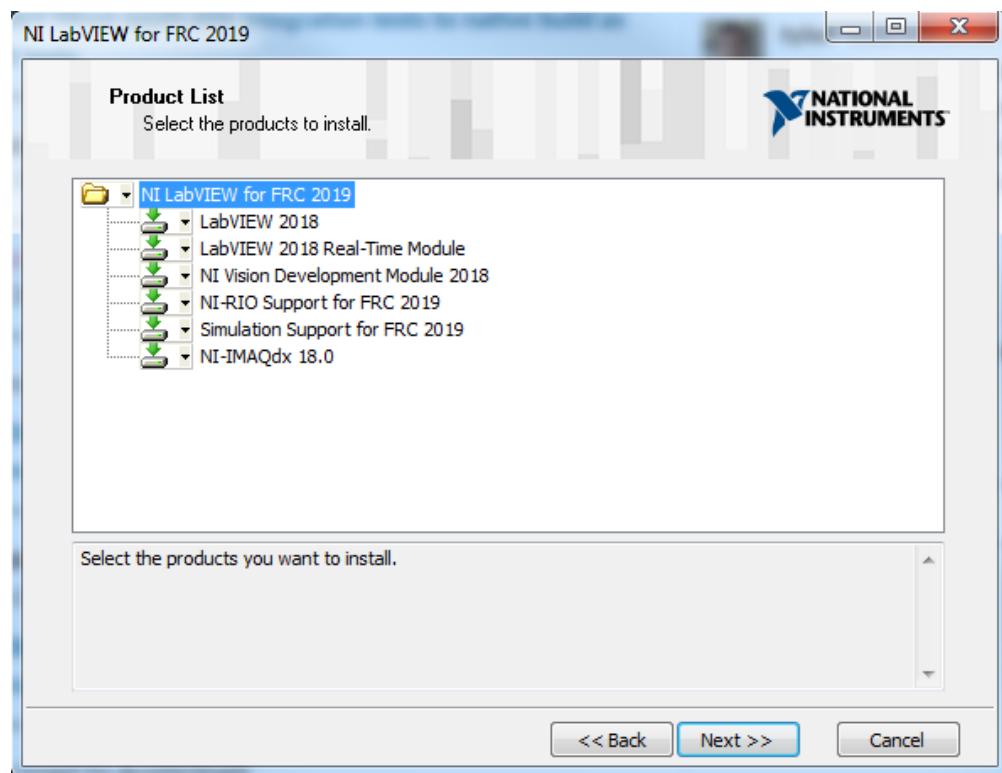
Double click on autorun.exe to launch the installer. If prompted to allow changes click Yes. To install LabVIEW to program your FRC robot, click the top option **Install Everything for LabVIEW Development**. To install only NI Vision Assistant for use with C++ or Java, click **Install Only NI Vision Development Module**. If prompted with any Windows security warnings, click Allow or Yes.

Warnings



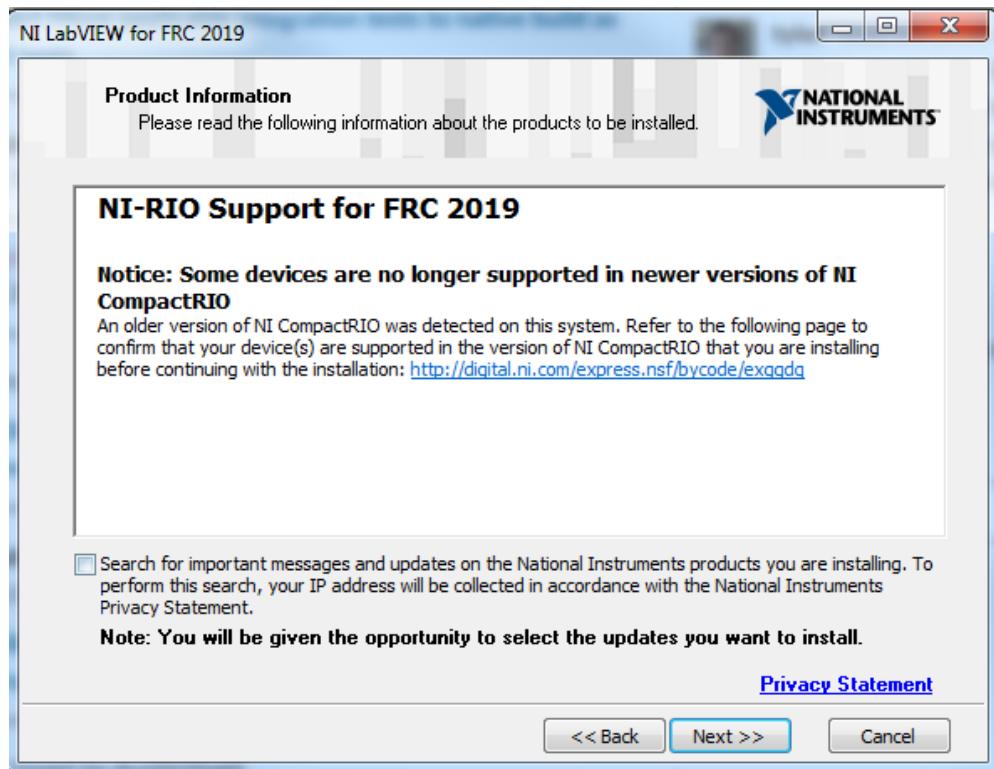
Click "Next"

Product List



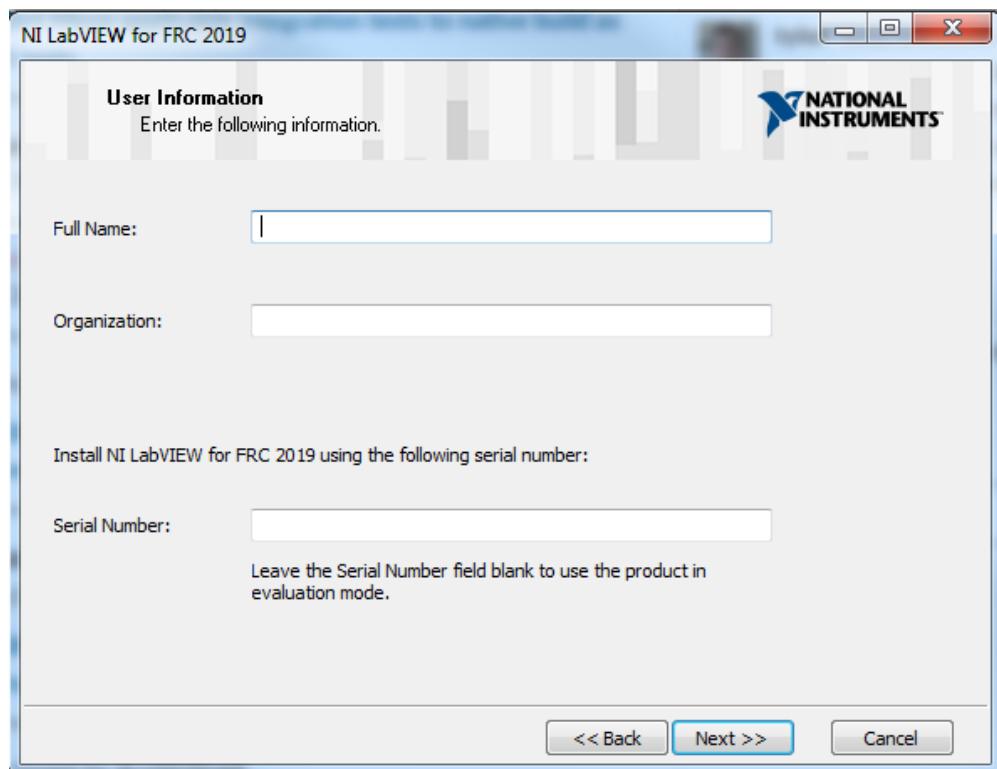
Click "Next"

Product Information



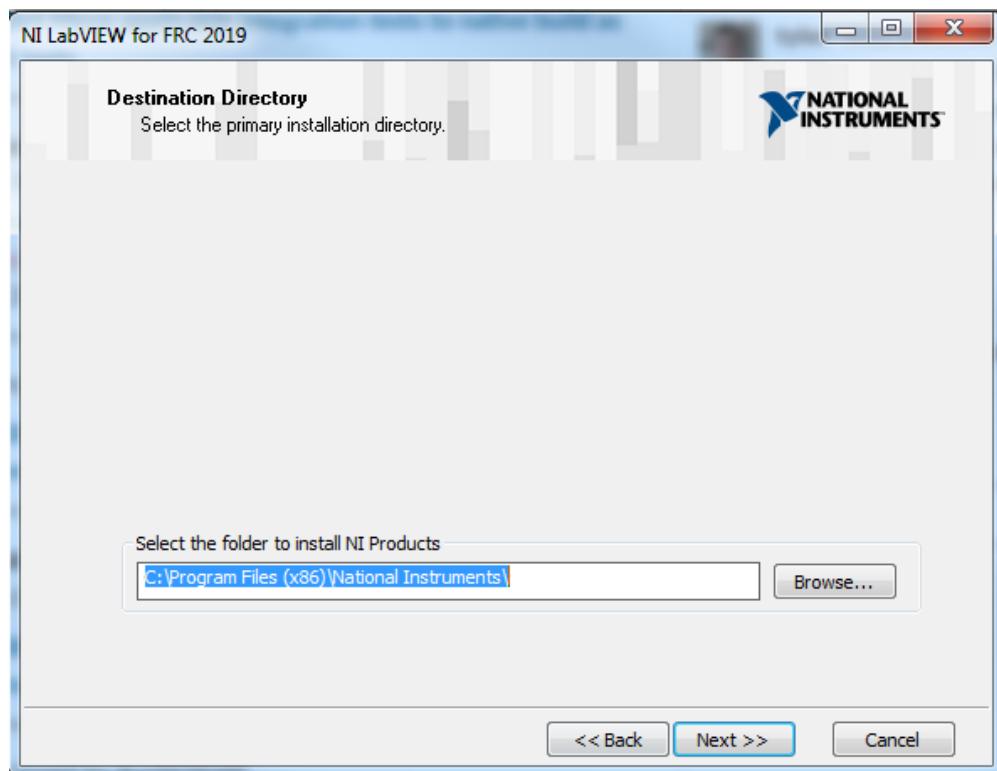
Un-check the box, then click "Next". (Note: you may not see the warning shown in the top portion of the window in this picture).

User Information



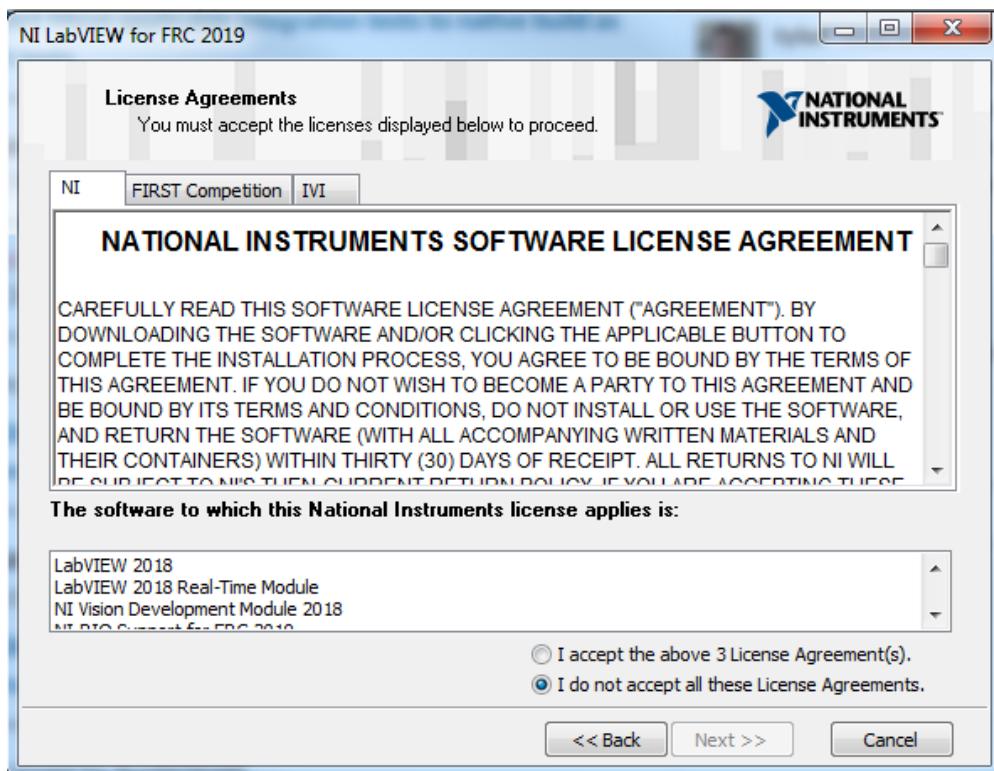
Enter name, organization, and the serial number from the [ReadMe in the File Releases on Teamforge](#). Click "Next". If you cannot find your serial number, please reach out to National Instruments at www.ni.com/frc/needhelp.

Destination Directory



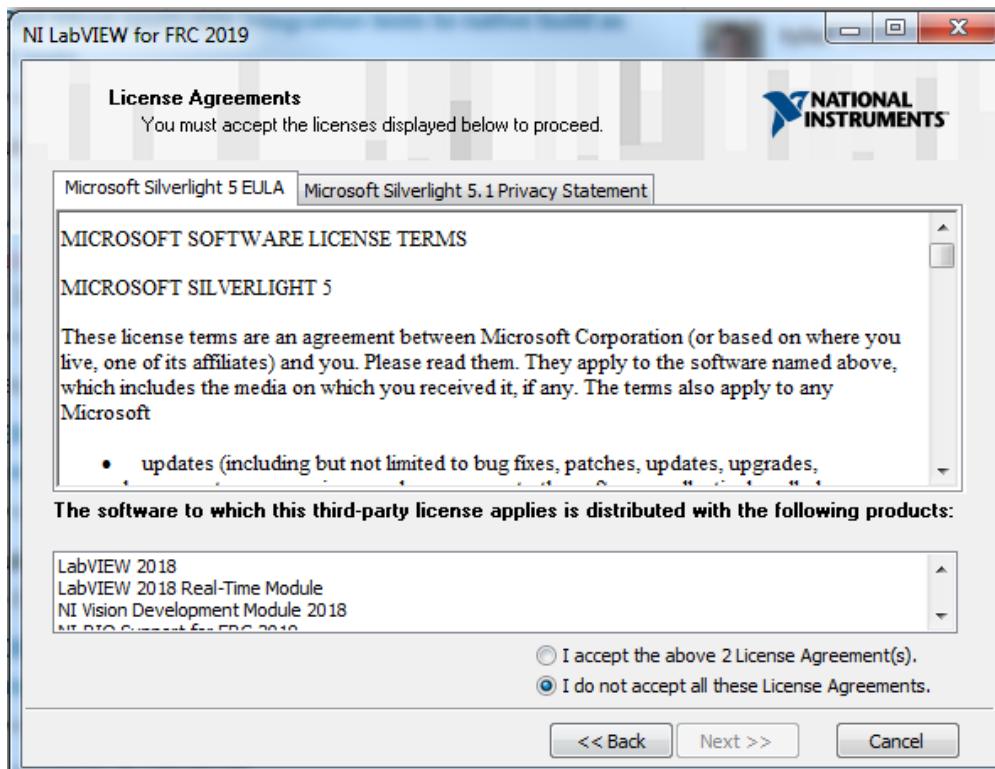
Click "Next"

License Agreements (1)



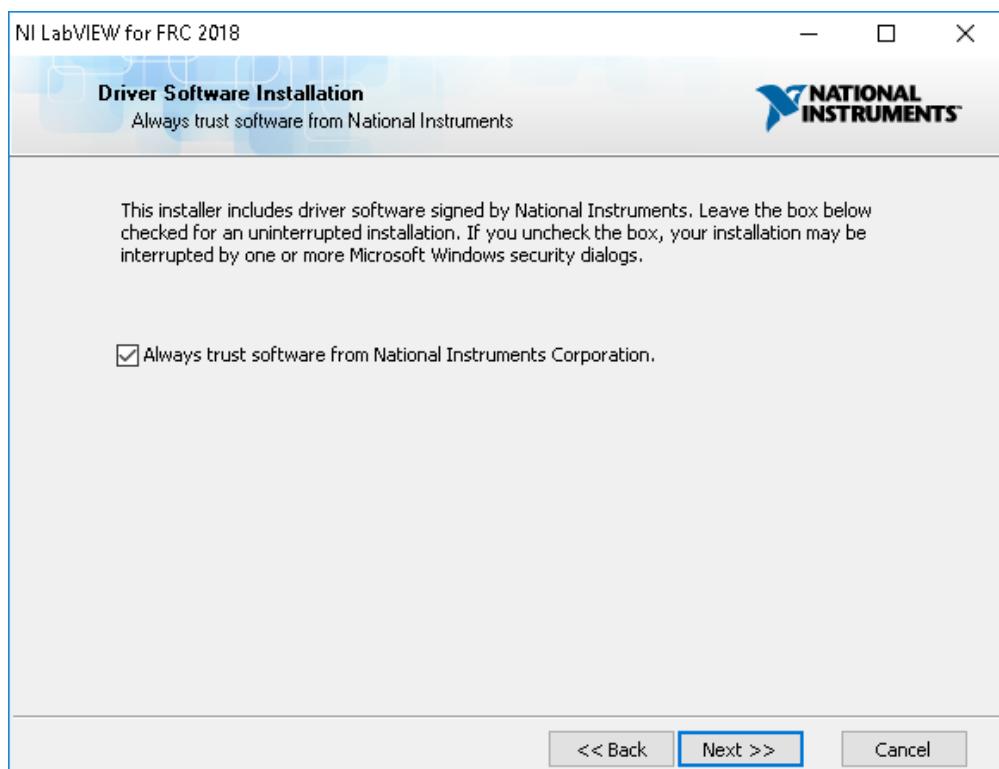
Check "I accept..." then Click "Next"

License Agreements (2)



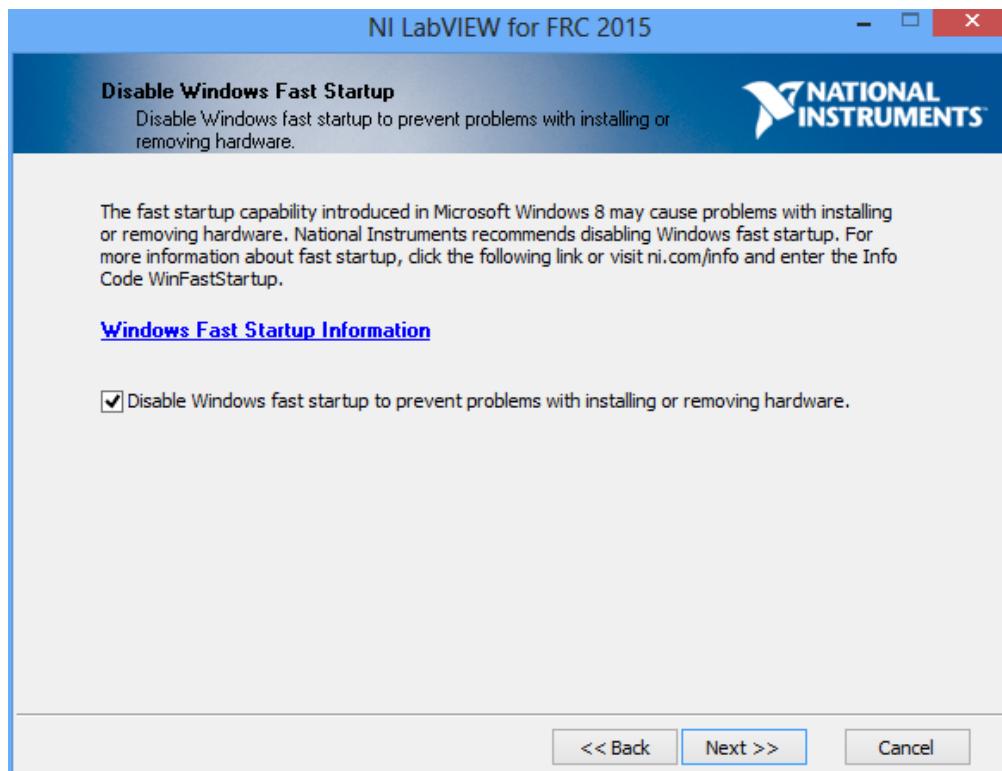
Check "I accept..." then Click "Next"

Driver Software Installation



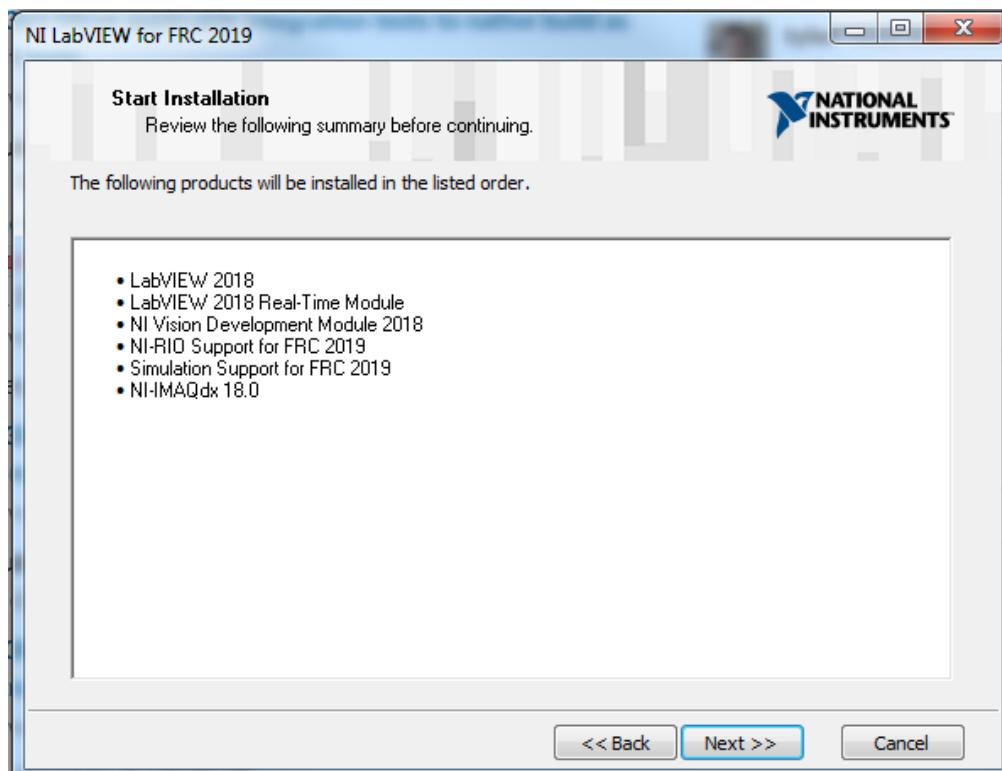
If you see this screen, Click "Next"

Disable Windows Fast Startup



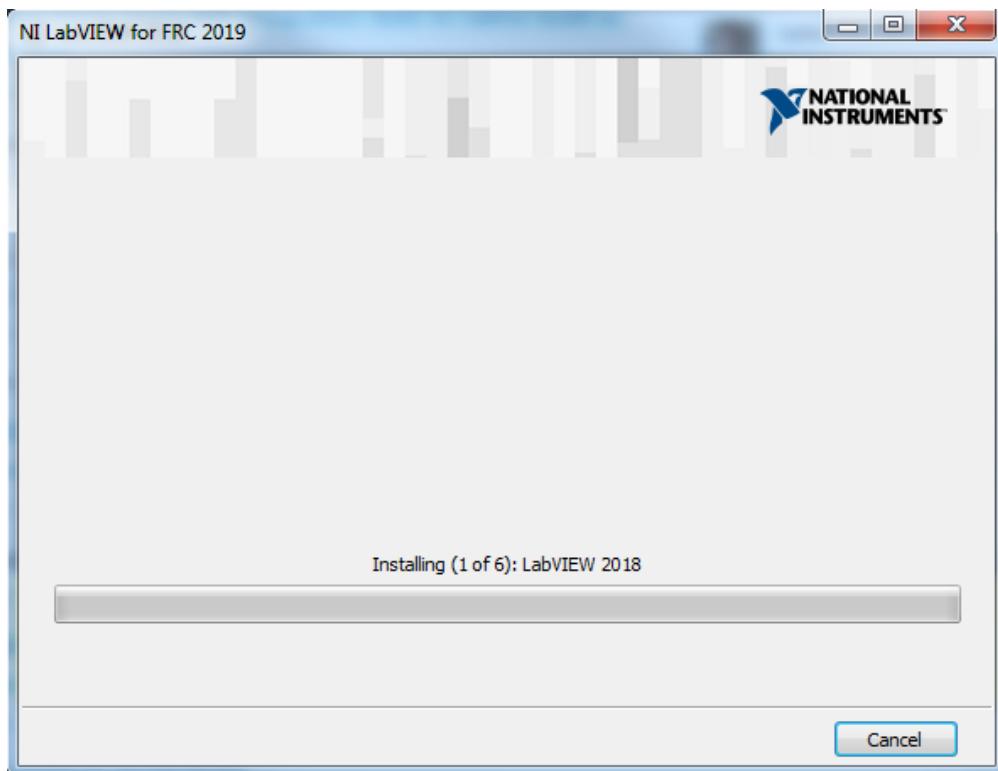
If you see this screen, click "Next"

Start Installation



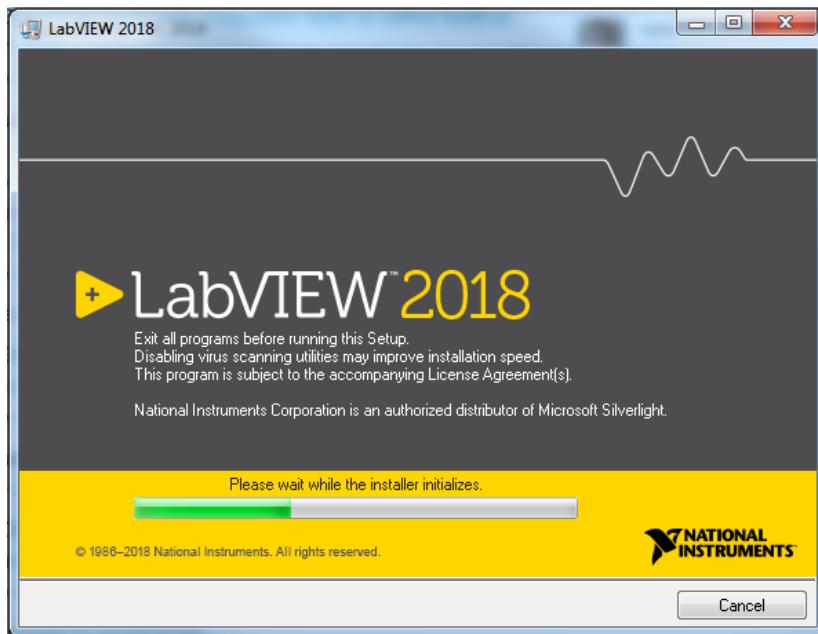
Click "Next"

Overall Progress



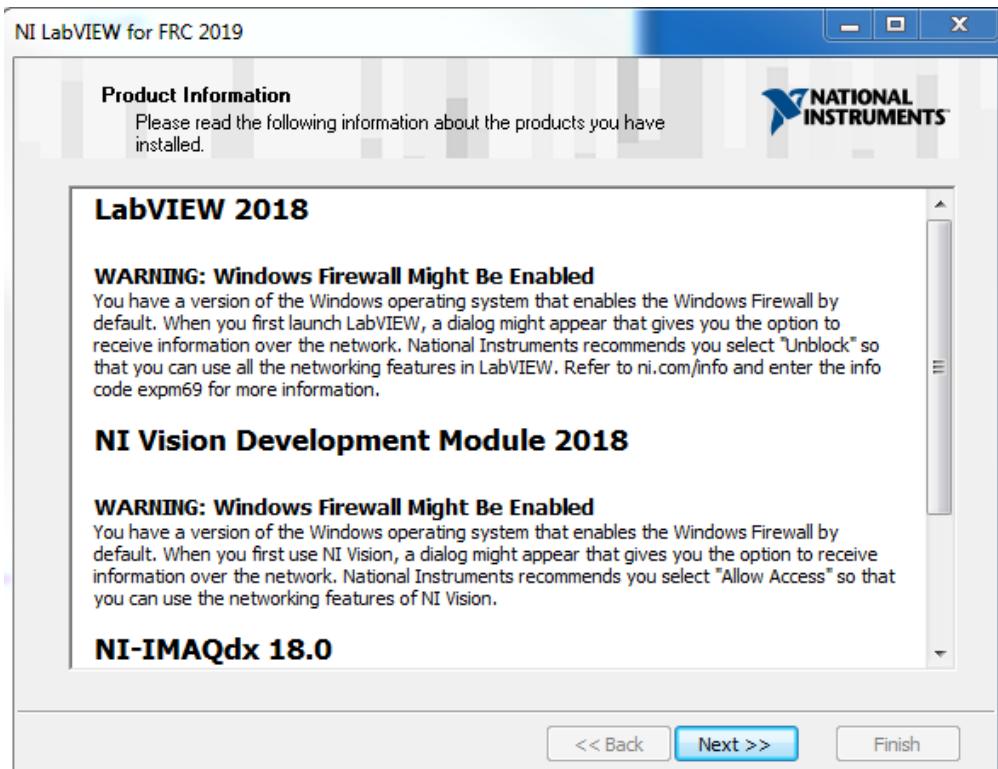
Overall installation progress will be tracked in this window

Individual Product Progress



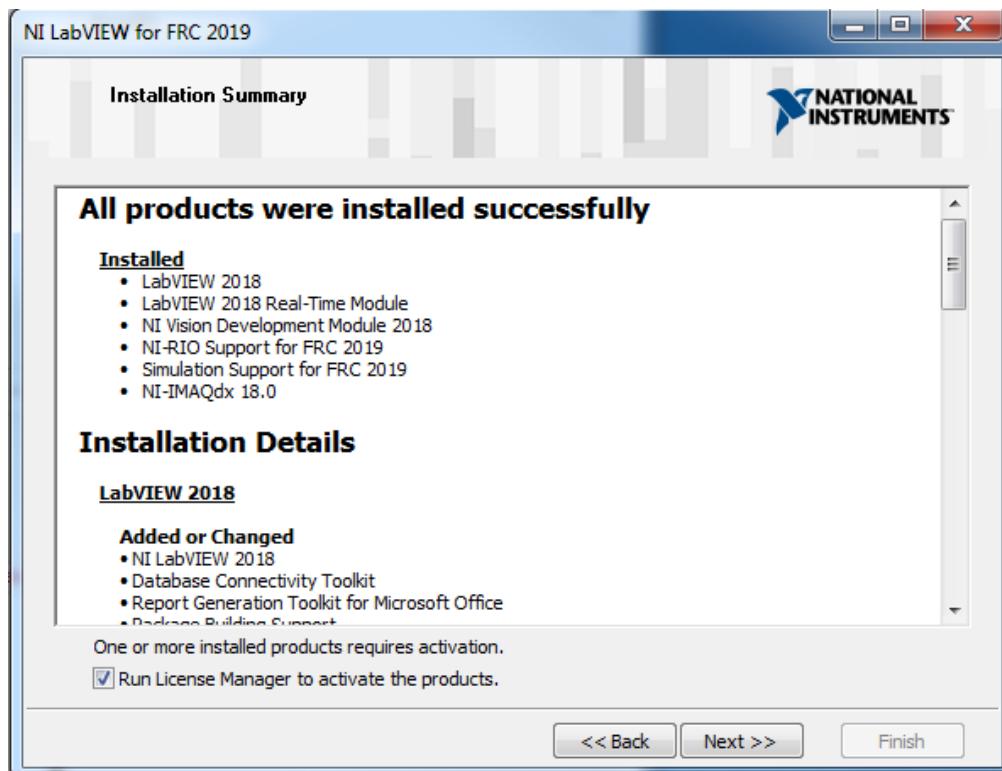
Each product installed will also create an individual progress window like the one shown above.

Product Information



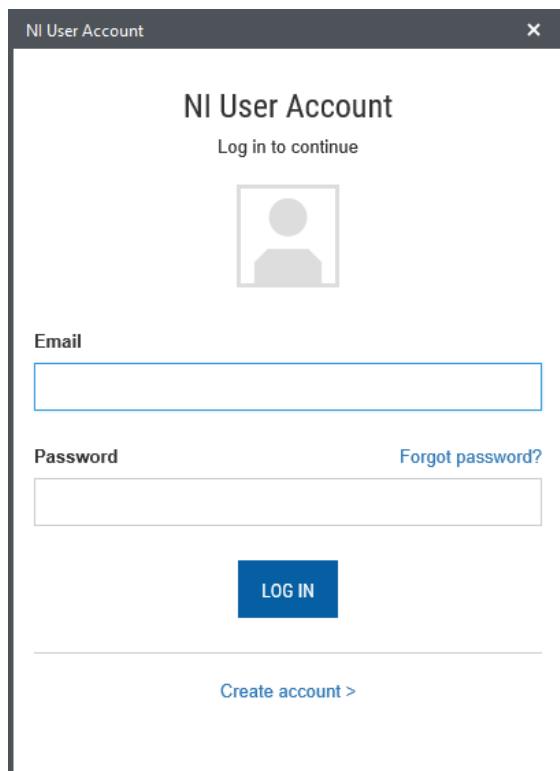
Click "Next"

Installation Summary



If internet access is available and you are ready to activate, click "Next"; otherwise uncheck the "Run License Manager..." and click "Next".

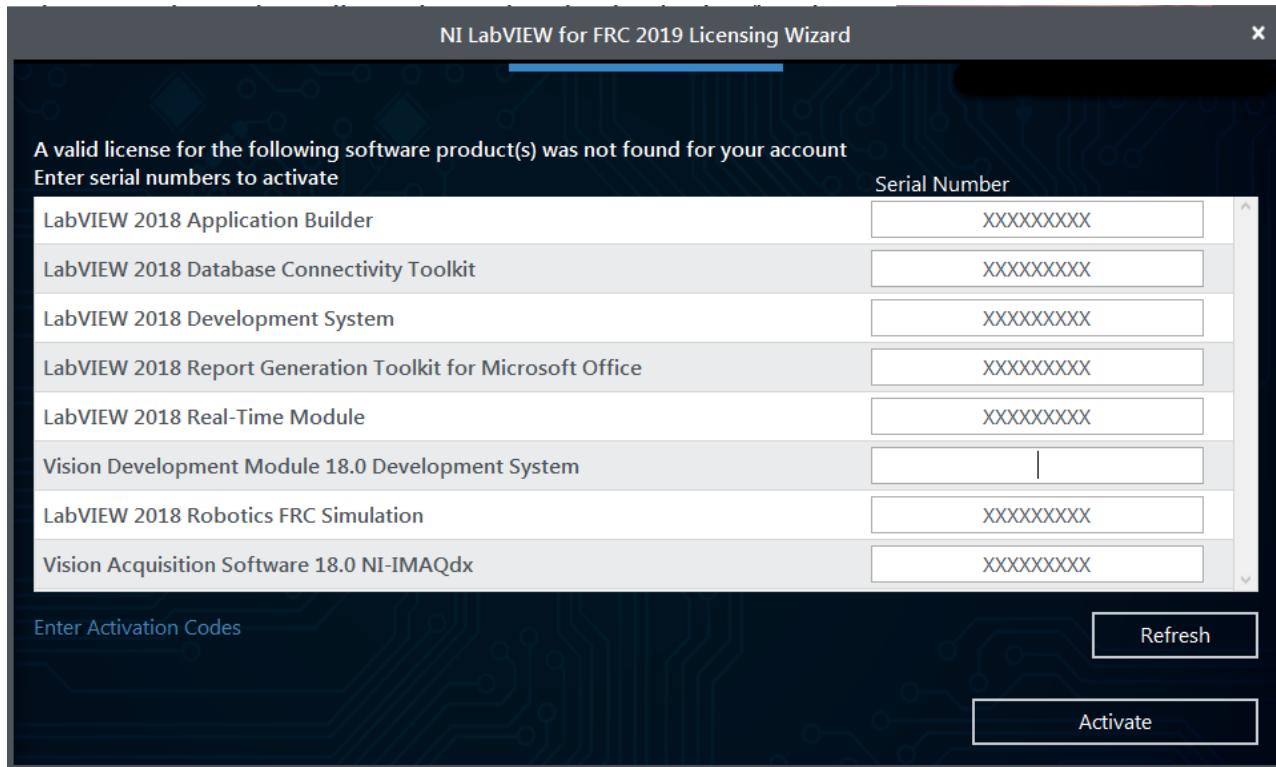
NI Activation Wizard



Log into your ni.com account. If you don't have an account, select 'Create account' to create a free account.

FRC 2019 Beta

NI Activation Wizard (2)

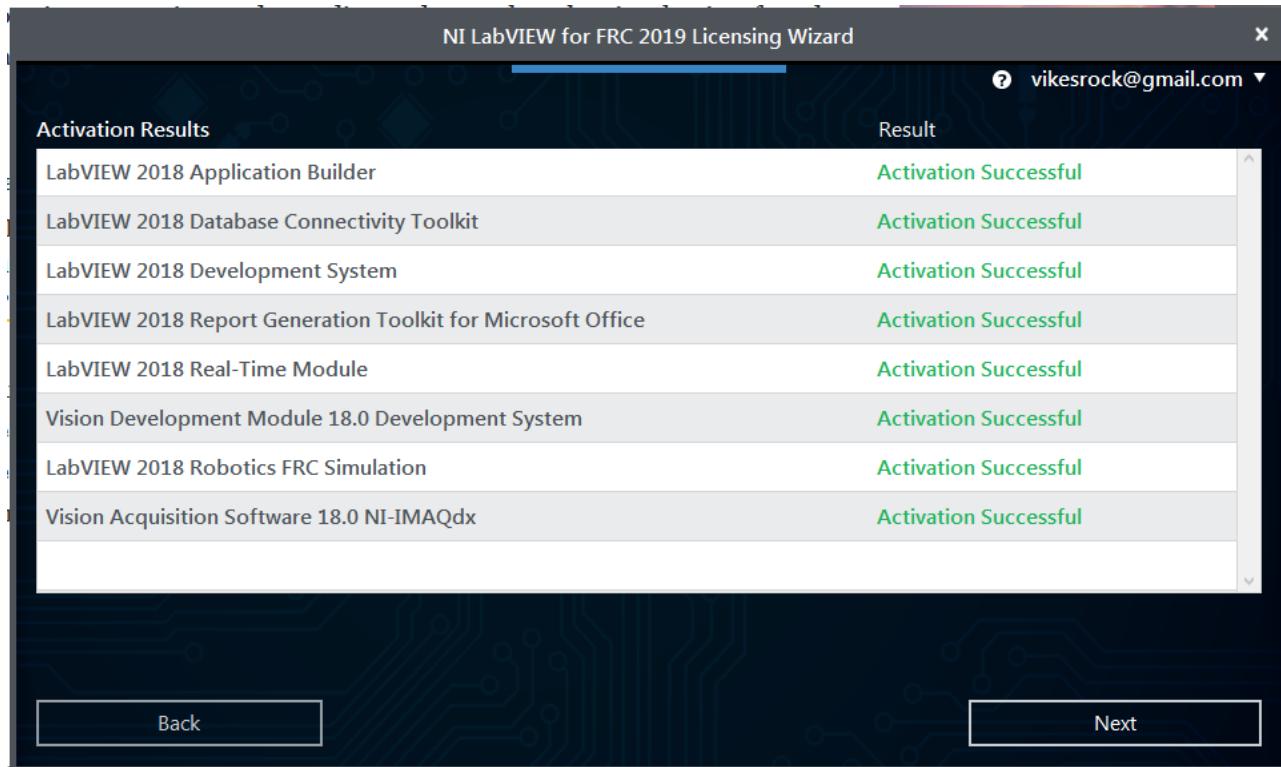


The serial number you entered at the "User Information" screen should appear in all of the text boxes, if it doesn't, enter it now. Click "Activate".

Note: If this is the first time activating the 2019 software on this account, you will see the message shown above about a valid license not being found. You can ignore this.

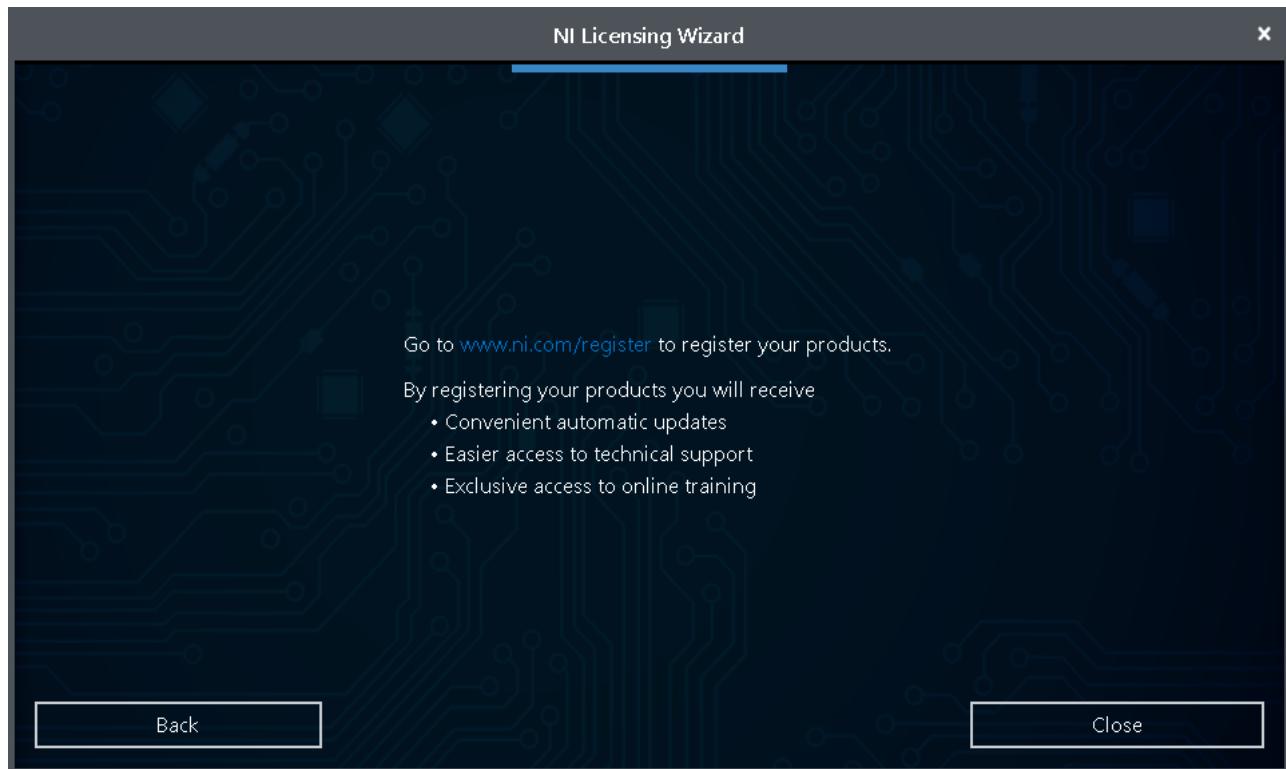
FRC 2019 Beta

NI Activation Wizard (3)



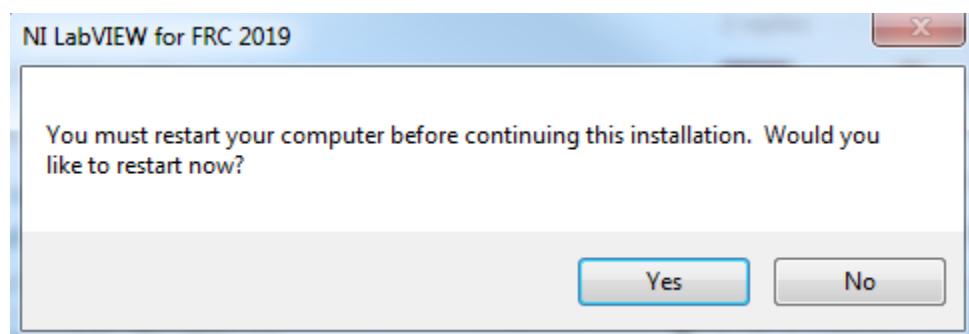
If your products activate successfully, an "Activation Successful" message will appear. If the serial number was incorrect, it will give you a text box and you can re-enter the number and select "Try Again". If everything activated successfully, click "Next".

NI Activation Wizard (4)



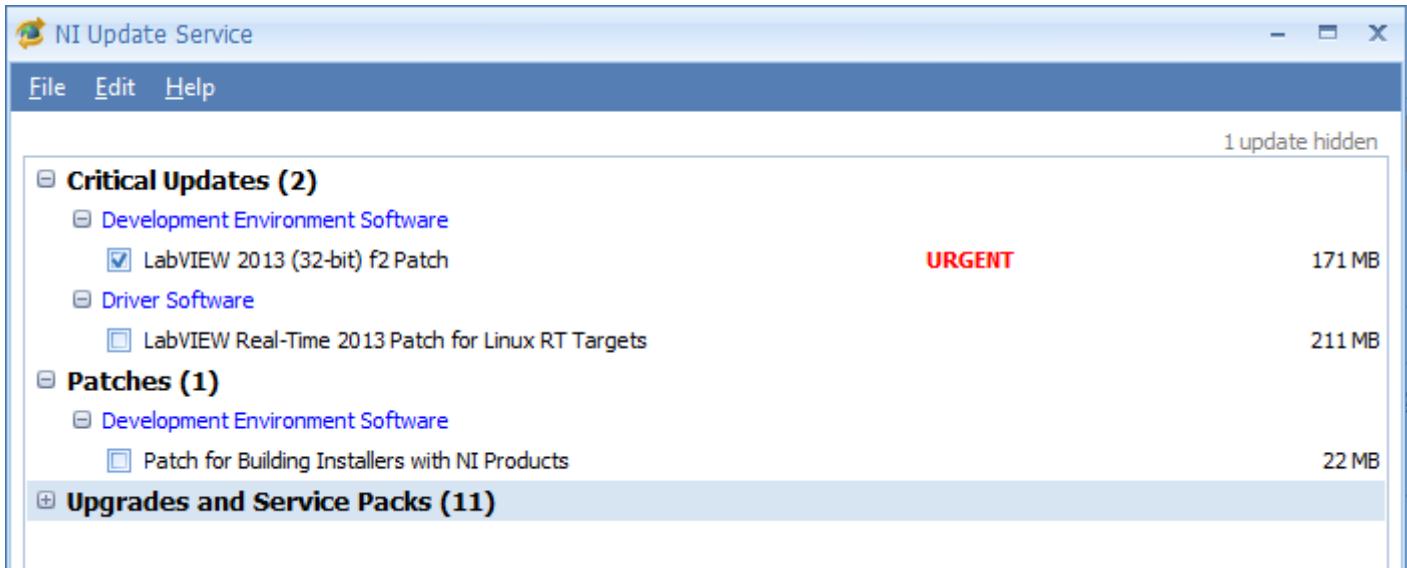
Click "Close".

Restart Message



Select "Yes"

NI Update Service



On occasion you may see alerts from the NI Update Service about patches to LabVIEW. **It is not recommended to install these updates unless directed by FRC through our usual communication channels (Frank's Blog, Team Updates or E-mail Blasts).**

Installing the FRC Update Suite (All Languages)

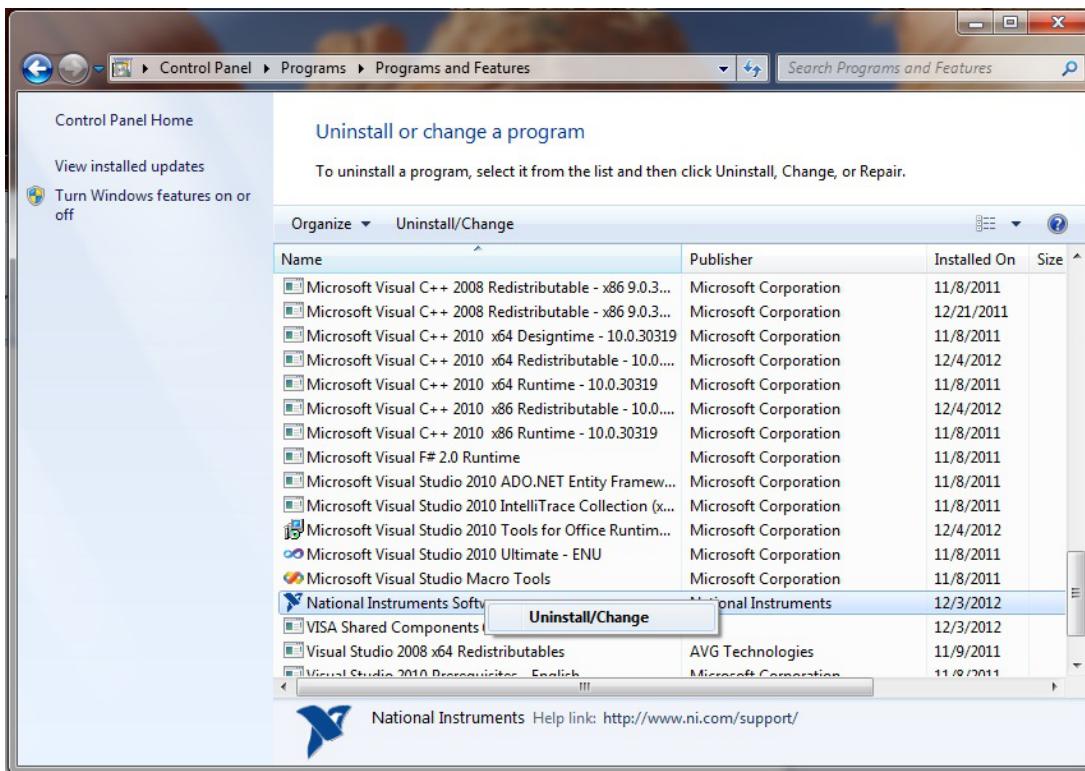


The FRC Update Suite contains the following software components: LabVIEW Update, FRC Driver Station, and FRC Utilities. If an FRC LabVIEW installation is found, the LabVIEW Update will be installed or updated, otherwise this step will be skipped. The FRC Driver Station and FRC Utilities will always be installed or updated. The LabVIEW runtime components required for the driver station and utilities is included in this package. No components from the LabVIEW Merged Suite are required for running either the Driver Station or Utilities.

C++ and Java teams wishing to use NI Vision Assistant should run the full Suite installer as described in the article - [Installing LabVIEW for FRC \(LabVIEW only\)](#)

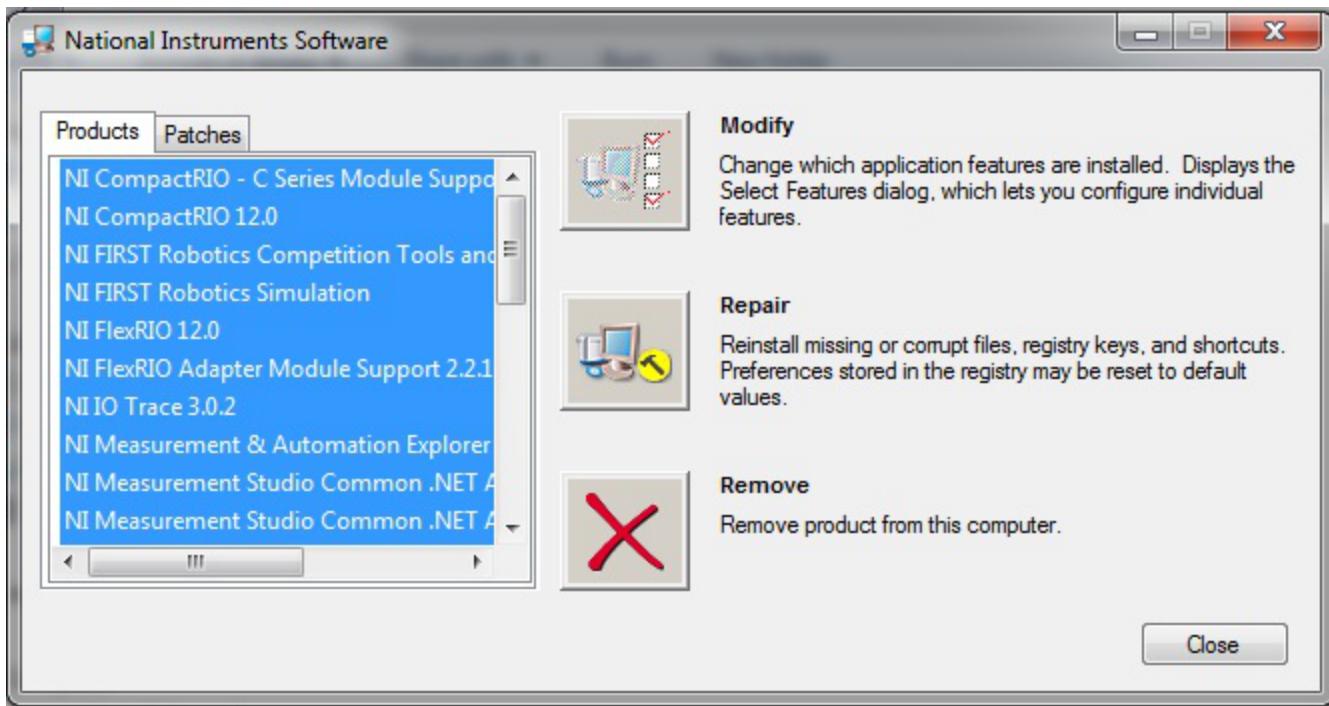
! Note: The Driver Station will only work on Windows 7, Windows 8, Windows 8.1, and Windows 10. It will not work on Windows XP.

Uninstall Old Versions (Recommended)



LabVIEW teams have already completed this step, do not repeat it. Before installing the new version of the NI Update it is recommended to remove any old versions. The new version will likely properly overwrite the old version, but all testing has been done with FRC 2019 only. Make sure to back up any team code located in the "User\LabVIEW Data" directory before un-installing. Then click Start >> Control Panel >> Uninstall a Program. Locate the entry labeled "National Instruments Software", right-click on it and select Uninstall/Change.

Select Components to Uninstall



In the left pane of the dialog box that appears, select all entries. The easiest way to do this is to click the top entry to highlight it, then scroll down to the bottom entry, press and hold shift and click on the last entry then release shift. Click Remove. Wait for the uninstaller to complete and reboot if prompted.

Downloading the Update

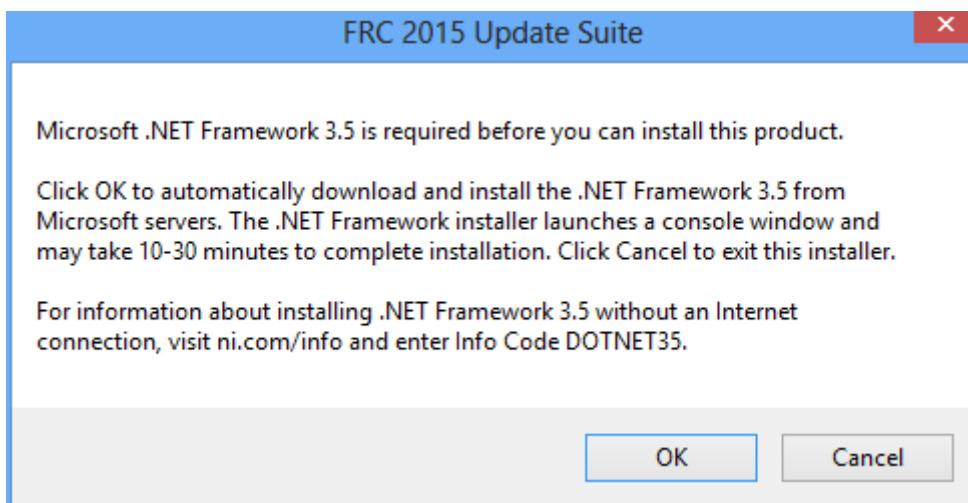
Download the update from <http://www.ni.com/download/first-robotics-software-2017/7183/en/>

⚠ Note: This download will require the decryption key from the Kickoff broadcast

.NET Framework 4.6.2

The Update installer may prompt that .NET Framework 4.6.2 needs to be updated or installed. Follow prompts on-screen to complete the installation, including rebooting if requested. Then resume the installation of the NI FRC Update, restarting the installer if necessary.

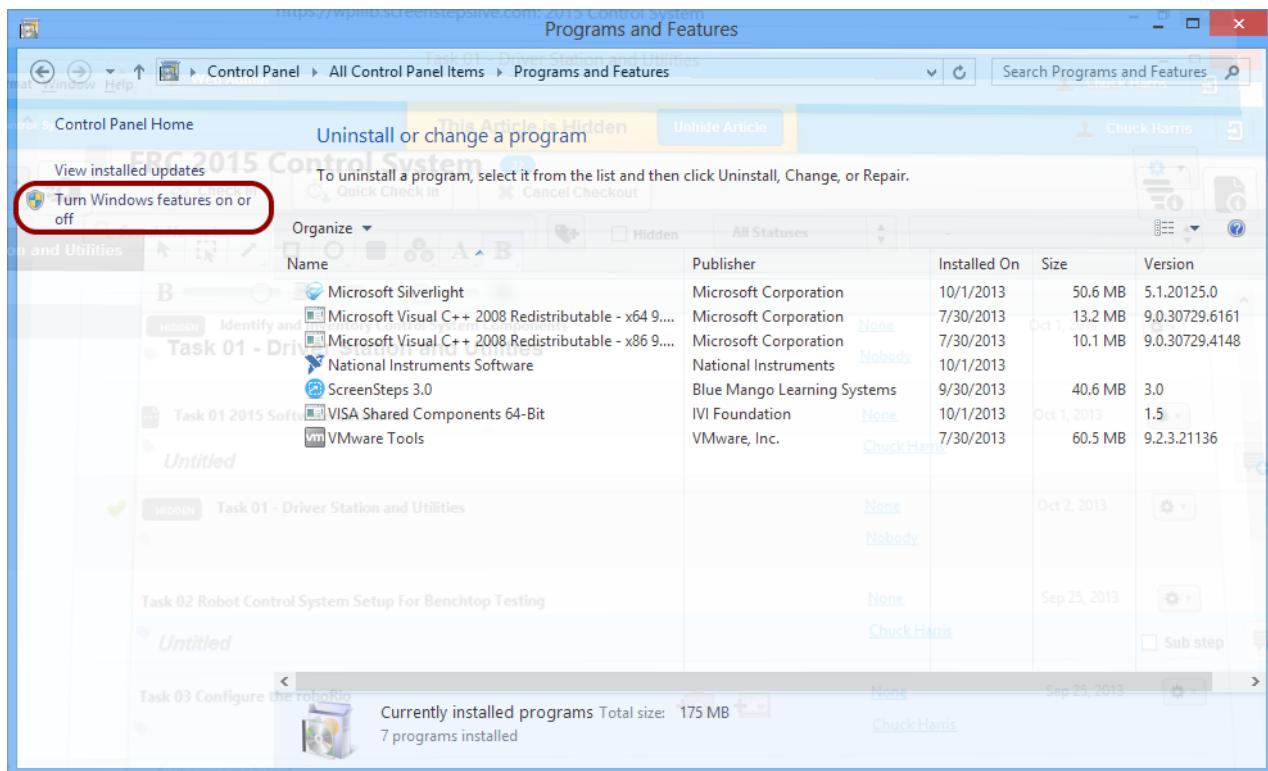
.NET Framework 3.5



If installing on Windows 8 or 10, the Microsoft .NET Framework 3.5 may need to be installed. If you see the dialog shown above, click "Cancel" and perform the steps shown below. An internet connection is required to complete these steps.

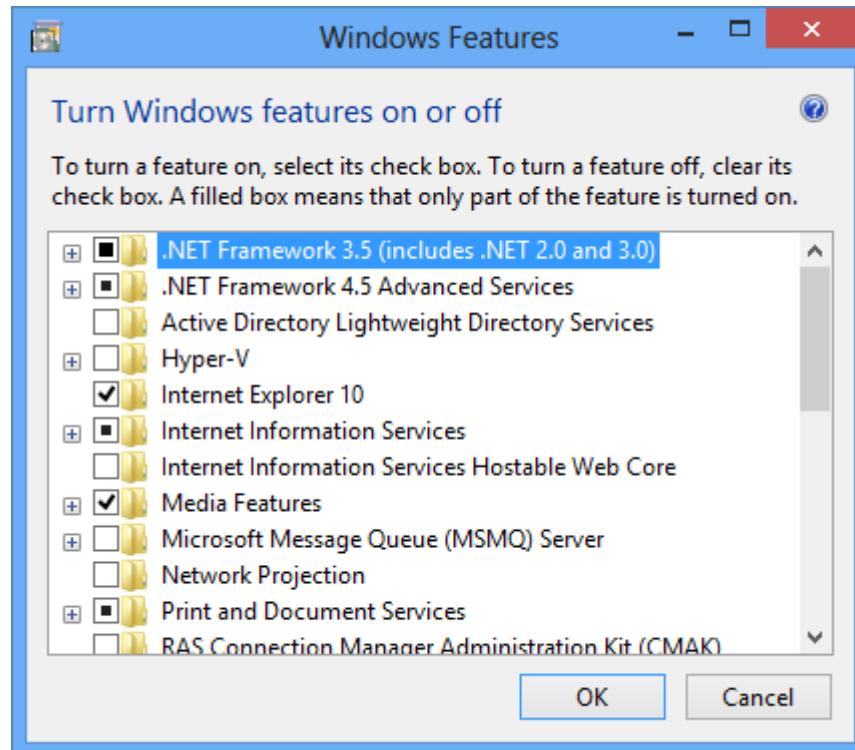
FRC 2019 Beta

Programs and Features



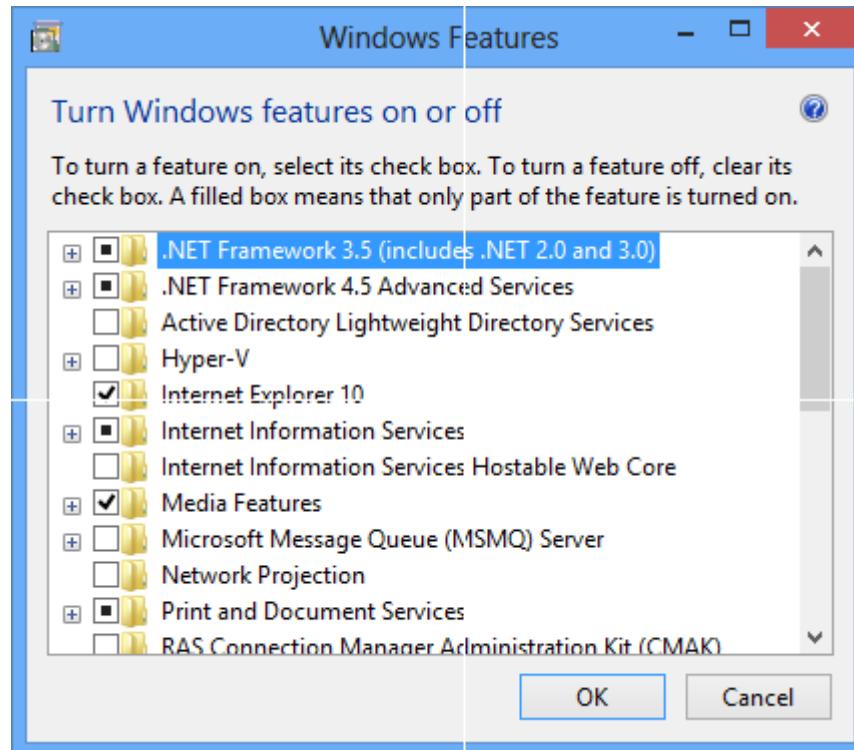
Open the "Programs and Features" window from the control panel and click on "Turn Windows features on or off"

Windows Features (.NET Framework 3.5 not on)



Select ".NET Framework 3.5 (includes .NET 2.0 and 3.0)" to enable it (a black dot, not a check box will appear) and then click "OK". When installation finishes [restart installation of FRC 2019 Update Suite](#).

Windows Features (.NET Framework 3.5 already on)



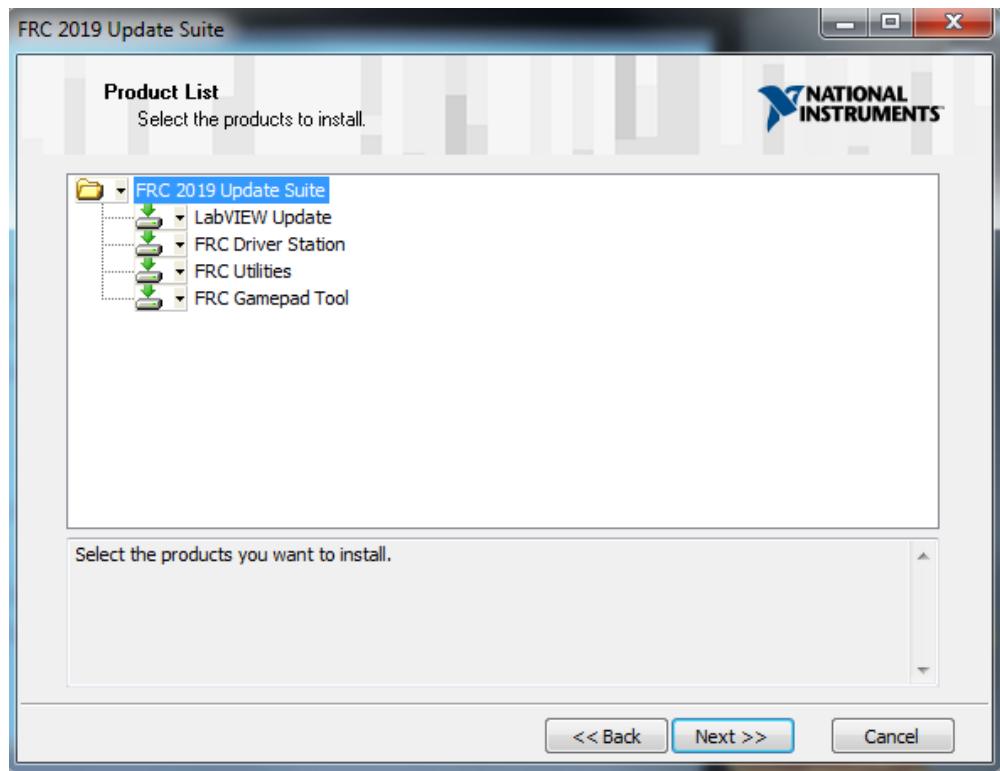
If a black dot is shown next to ".NET Framework 3.5" the feature is already on. Click "Cancel" and [restart installation of FRC 2019 Update Suite](#).

Welcome



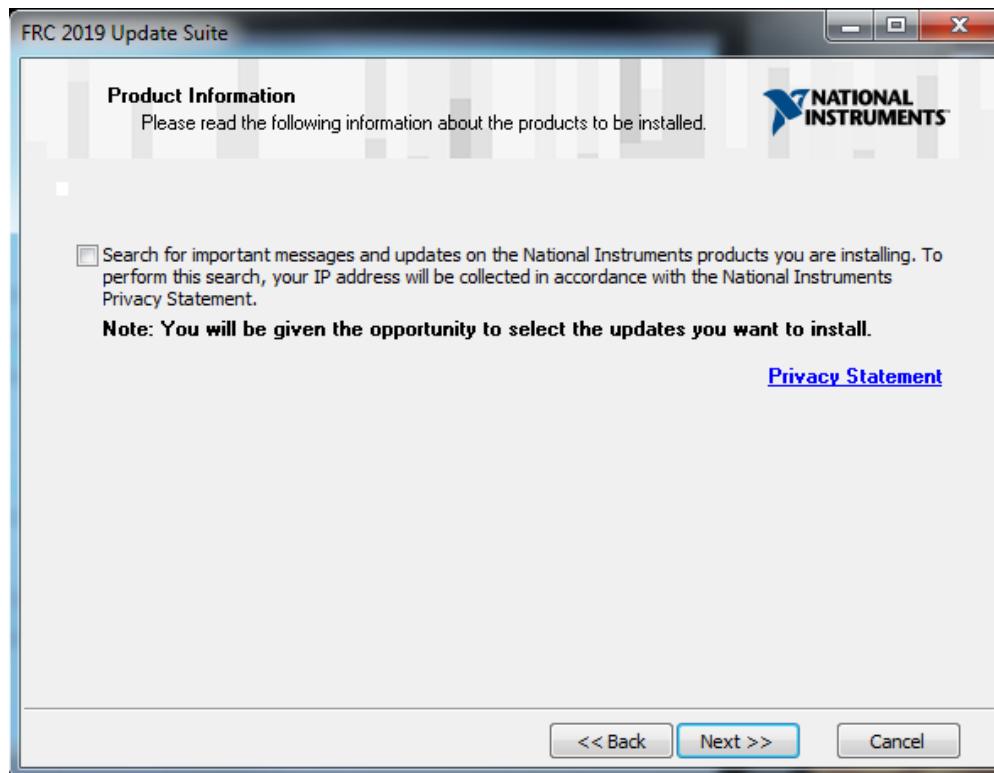
Right click on the downloaded zip file and select Extract All. If you downloaded the encrypted zip file, you will be prompted for the encryption key which will be released at Kickoff. Open the extracted folder and any subfolders until you reach the folder containing "setup" (may say "setup.exe" on some machines). Double click on the setup icon to launch the installer. Click "Yes" if a Windows Security prompt appears. Click "Next" on the splash screen that appears.

Product List



Click "Next". There is no need to de-select "LabVIEW Update" for C++ or Java teams, if you do not have the base LabVIEW installation (because you are not programming in LabVIEW) this installation will be skipped automatically.

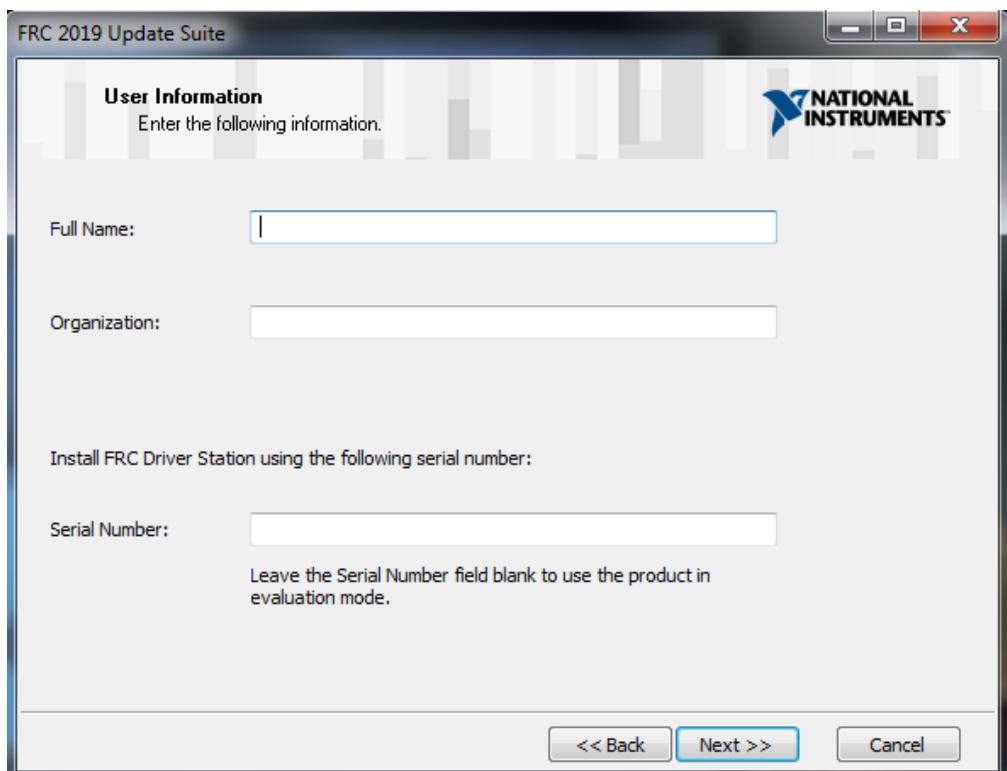
Product Information



Un-check the box, then Click "Next".

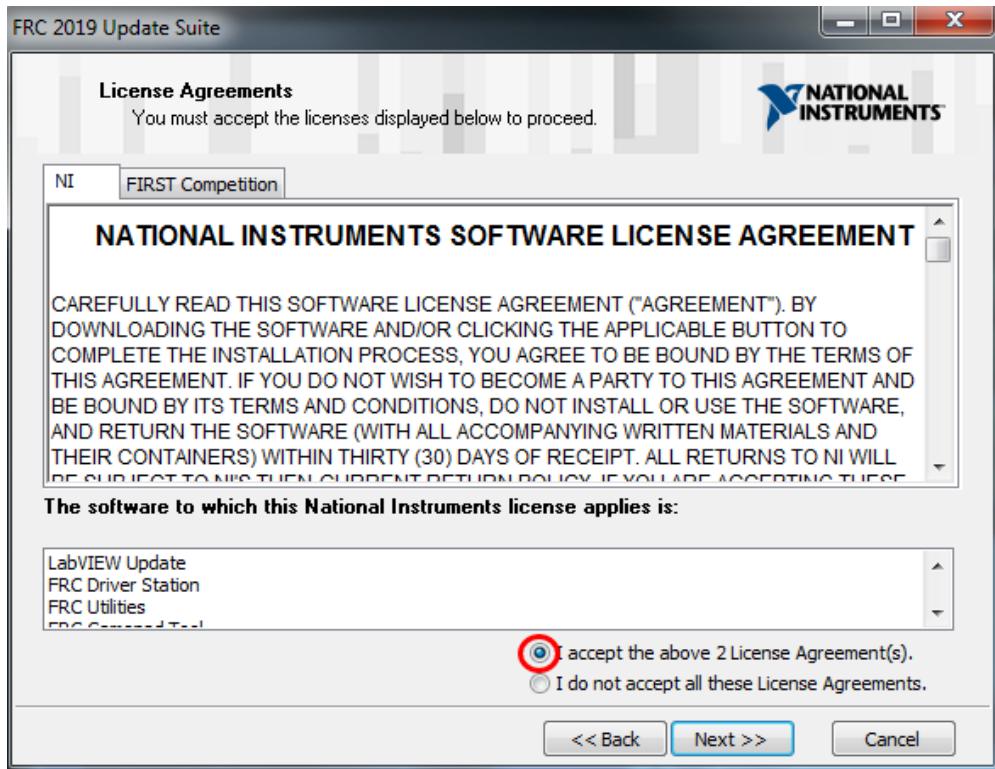
If you see a screen asking to disable Windows Fast Startup, leave it at the recommended option (disable Fast Startup) and click Next.

User Information



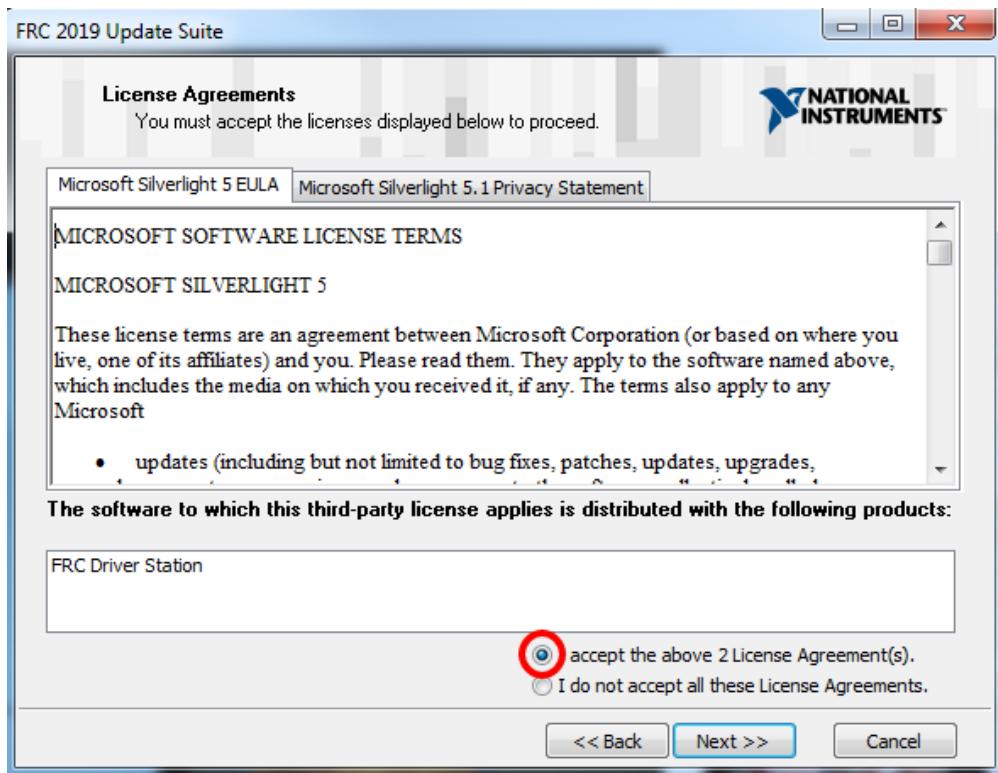
Enter full name and organization and the serial number from your kit of parts then click Next

License Agreements



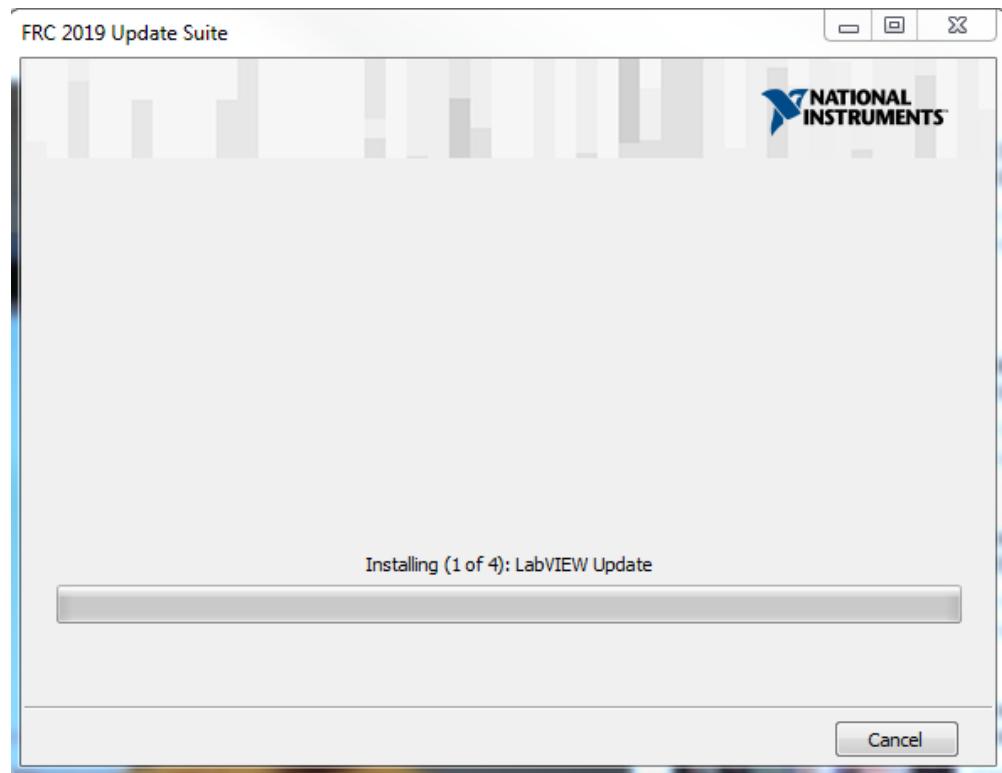
Select "I accept..." then click "Next"

License Agreements Page 2

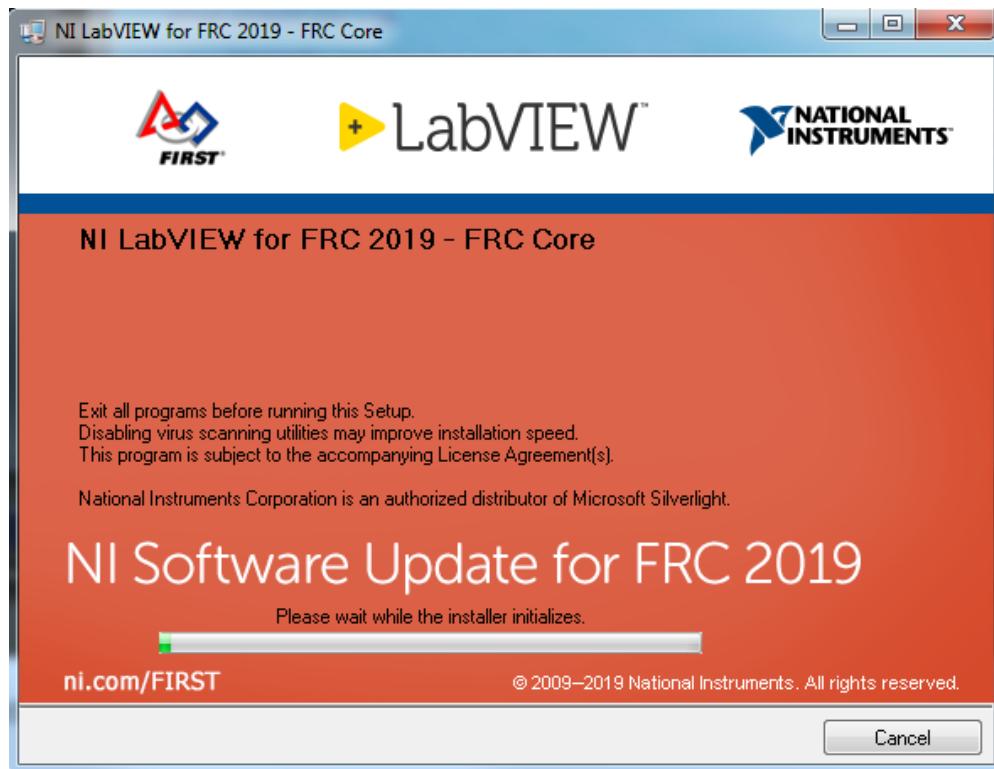


Select "I accept..." then click "Next"

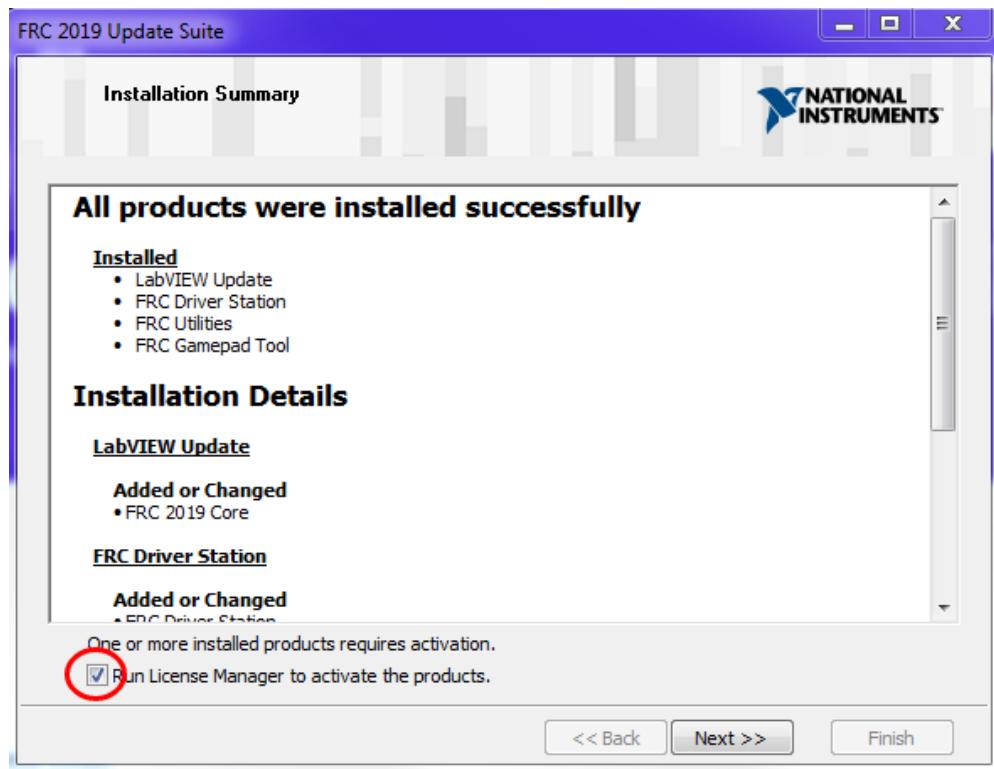
Summary Progress



Detail Progress

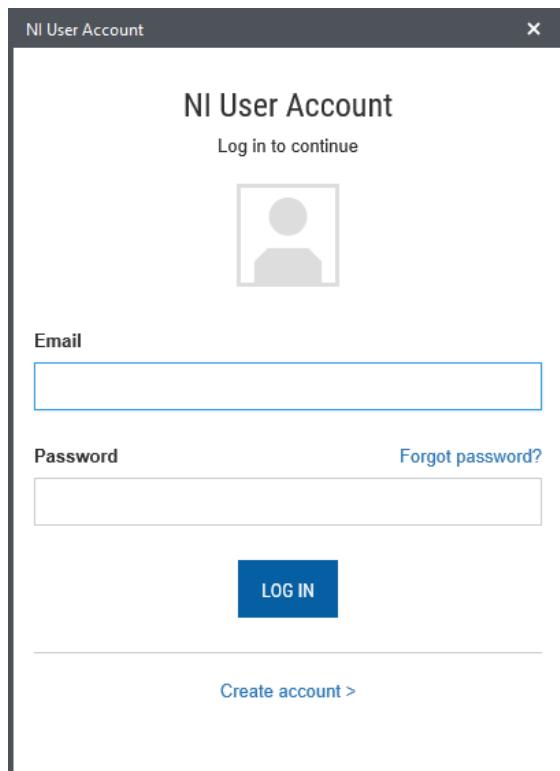


Installation Summary



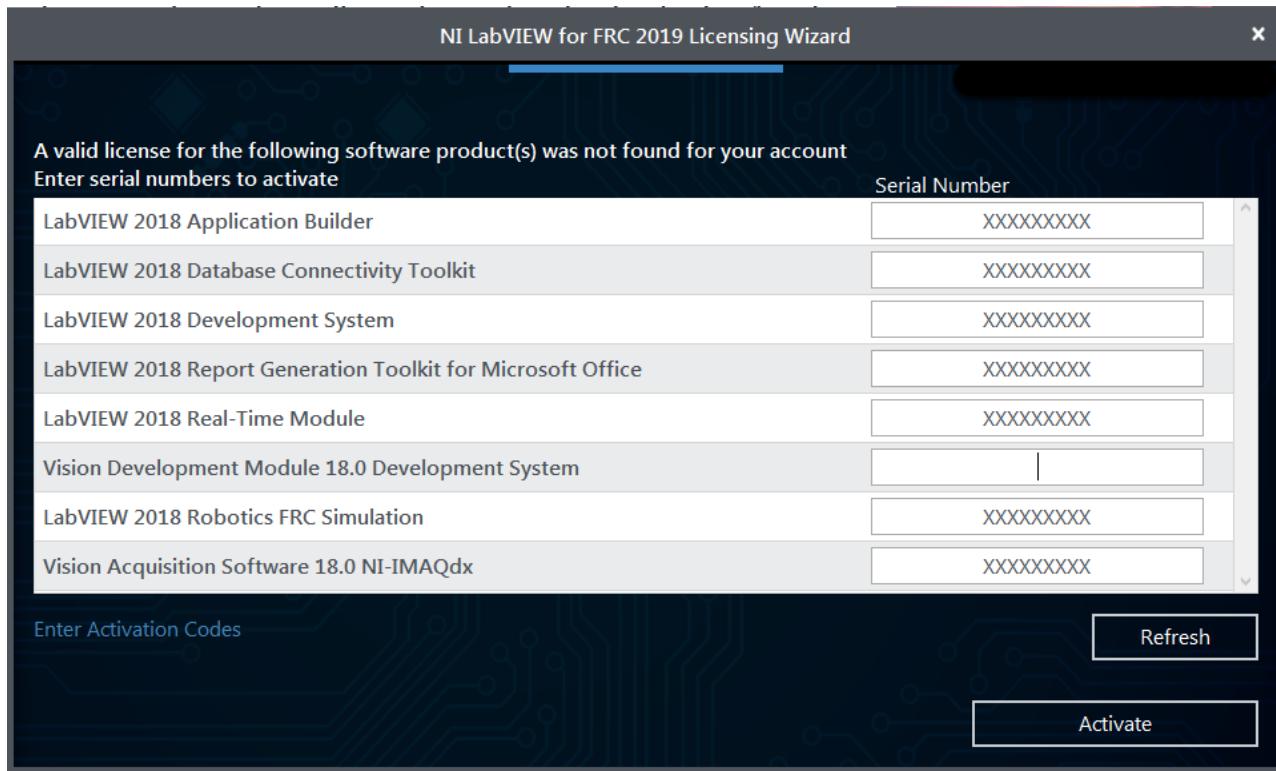
Make sure the box is checked to Run License Manager... then click Next

NI Activation Wizard



Log into your ni.com account. If you don't have an account, select 'Create account' to create a free account.

NI Activation Wizard (2)



The serial number you entered at the "User Information" screen should appear in all of the text boxes, if it doesn't, enter it now. Click "Activate".

Note: If this is the first time activating the 2019 software on this account, you will see the message shown above about a valid license not being found. You can ignore this.

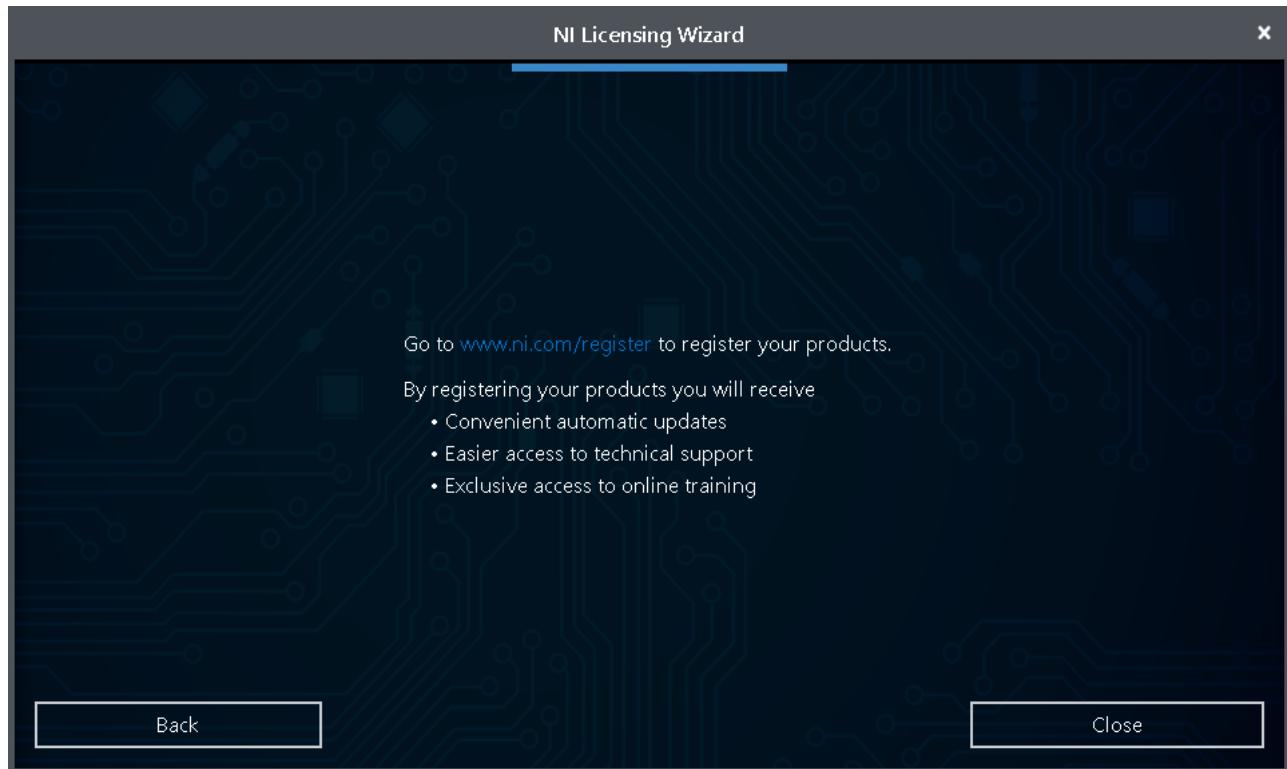
FRC 2019 Beta

NI Activation Wizard (3)



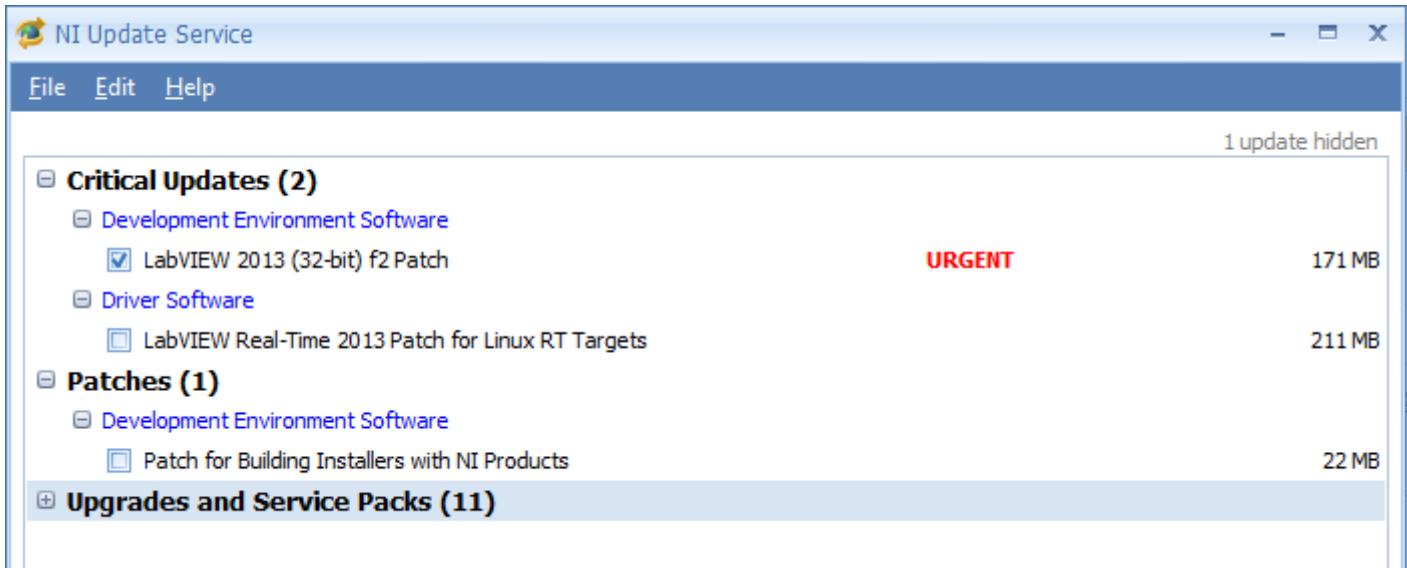
If your products activate successfully, an “Activation Successful” message will appear. If the serial number was incorrect, it will give you a text box and you can re-enter the number and select “Try Again”. If everything activated successfully, click “Next”.

NI Activation Wizard (4)



Click "Close".

NI Update Service



On occasion you may see alerts from the NI Update Service about patches to LabVIEW. It is not recommended to install these patches. FRC will communicate any recommended updates through our usual channels (Frank's Blog, Team Updates or E-mail Blasts).

Imaging your roboRIO

Before imaging your roboRIO, you must have completed installation of the [FRC Update Suite](#). You also must have the roboRIO power properly wired to the Power Distribution Panel as described [here](#).

Make sure the power wires to the roboRIO are secure and that the connector is secure firmly to the roboRIO (4 total screws to check).

Configuring the roboRIO

The roboRIO Imaging Tool will be used to image your roboRIO with the latest software.

USB Connection



FRC 2019 Beta

Connect a USB cable from the roboRIO USB Device port to the PC. This requires a USB Type A male (standard PC end) to Type B male cable (square with 2 cut corners), most commonly found as a printer USB cable.

Note: The roboRIO should only be imaged via the USB connection. It is not recommended to attempt imaging using the Ethernet connection.

Driver Installation

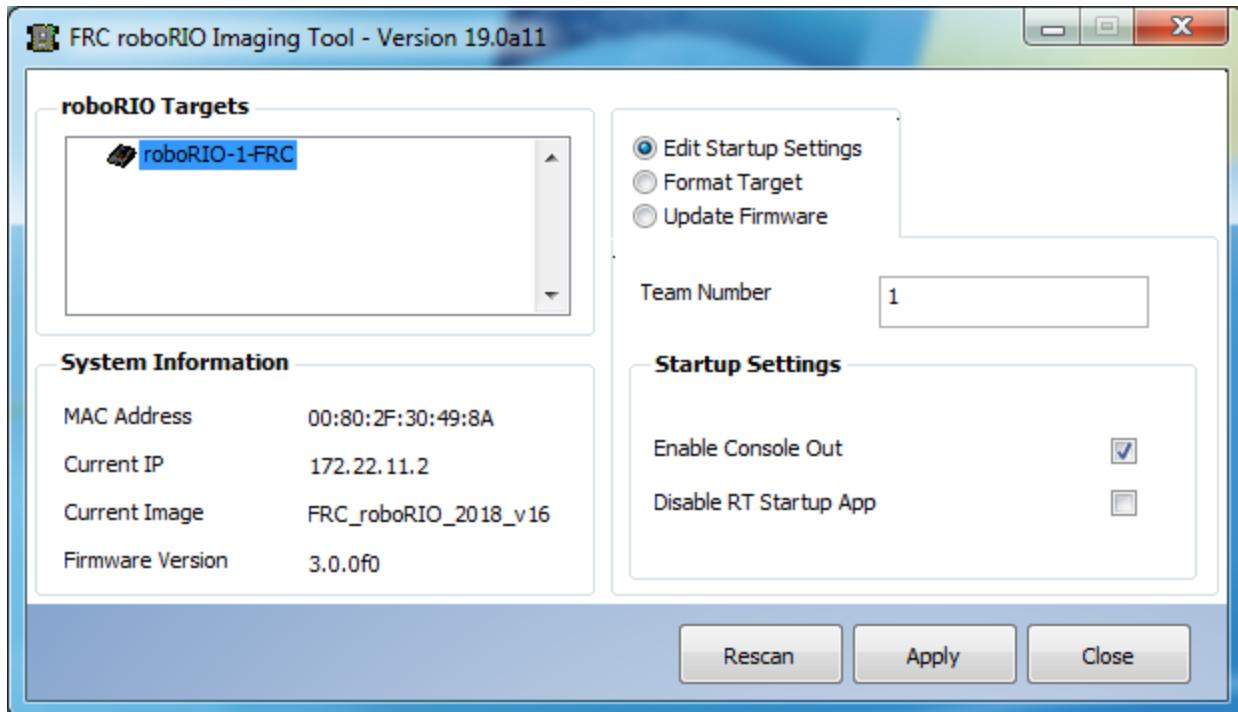
The device driver should install automatically. If you see a "New Device" pop-up in the bottom right of the screen, wait for the driver install to complete before continuing.

Launching the Imaging Tool



The roboRIO imaging tool and latest image are installed with the NI Update Suite. Launch the imaging tool by double clicking on the shortcut on the Desktop. If you have difficulties imaging your roboRIO, you may need to try right-clicking on the icon and selecting Run as Administrator instead.

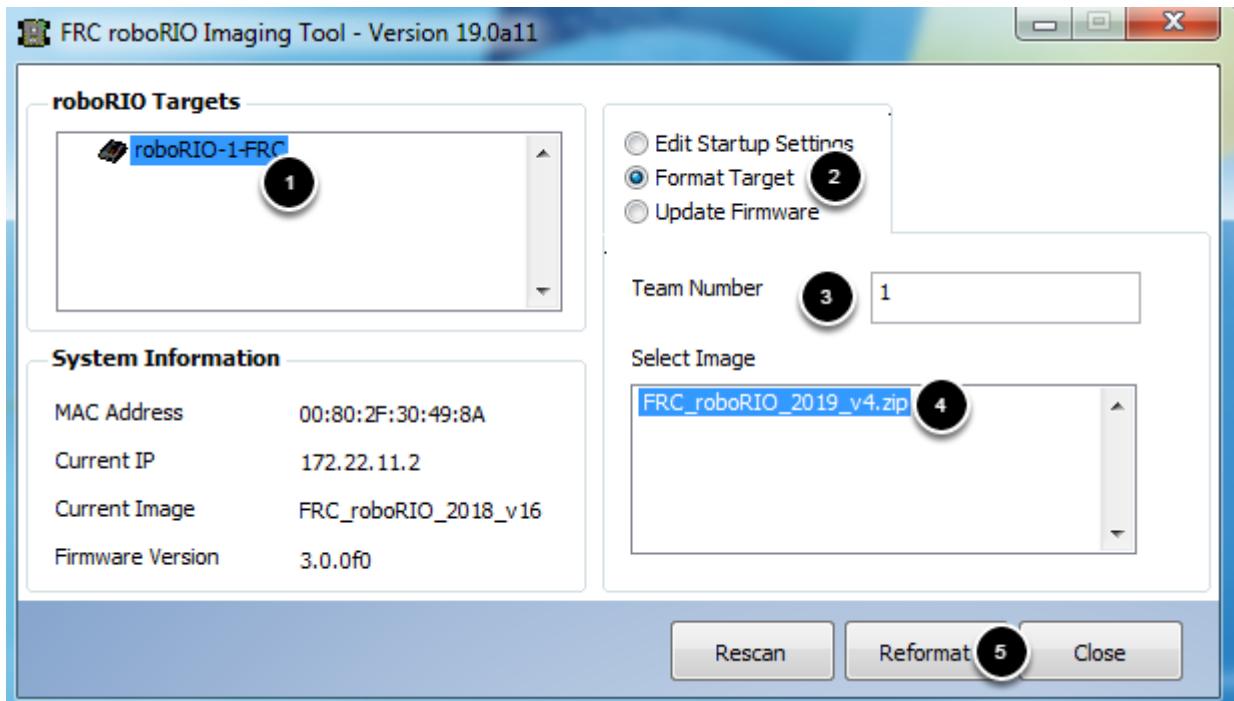
roboRIO Imaging Tool



After launching, the roboRIO Imaging Tool will scan for available roboRIOs and indicate any found in the top left box. The bottom left box will show information and settings for the roboRIO currently selected. The right hand pane contains controls for modifying the roboRIO settings:

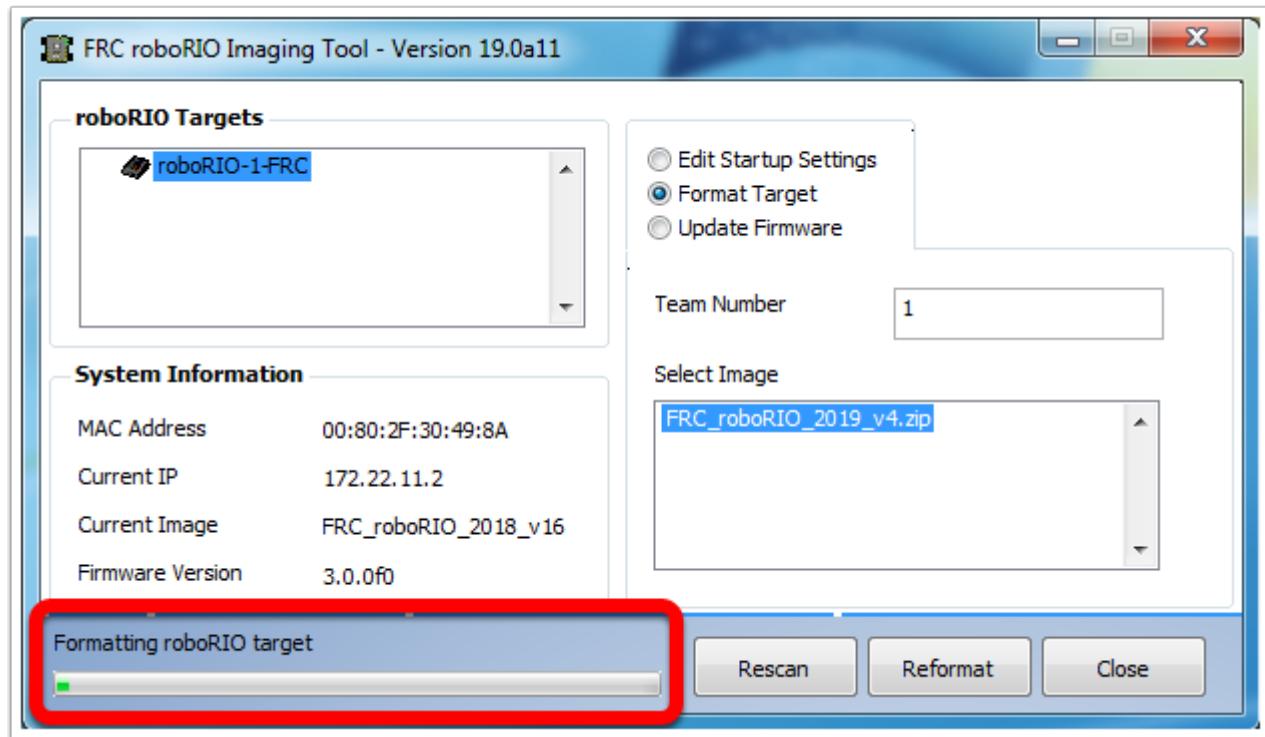
- **Edit Startup Settings** - This option is used when you want to configure the startup settings of the roboRIO (the settings in the right pane), without imaging the roboRIO.
- **Format Target** - This option is used when you want to load a new image on the roboRIO (or reflash the existing image). This is the most common option.
- **Update Firmware** - This option is used to [update the roboRIO firmware](#). As long as the roboRIO firmware is 3.0.0f0 or greater, there is likely no need to update.

Imaging the roboRIO



1. Make sure the roboRIO is selected in the top left pane
2. Select Format Target in the right pane
3. Enter your team number in the box
4. Select the latest image version in the box.
5. Click Reformat to begin the imaging process.

Imaging Progress



The imaging process will take approximately 3-10 minutes. A progress bar in the bottom left of the window will indicate progress.

Imaging Complete



When the imaging completes you should see the dialog above. Click Ok, then click the Close button at the bottom right to close the imaging tool. **Reboot the roboRIO using the Reset button to have the new team number take effect.**

 Note: The default CAN webdash functionality has been removed from the image (CAN devices will still work from robot code). You will need to use the tools provided by individual vendors to service their CAN devices.

Troubleshooting

If you are unable to image your roboRIO, troubleshooting steps include:

1. Try running the roboRIO Imaging Tool as Administrator by right-clicking on the Desktop icon to launch it.
2. Try accessing the roboRIO webpage with a web-browser at <http://172.22.11.2/> and/or verify that the NI network adapter appears in your list of Network Adapters in the Control Panel. If not, try re-installing the NI Update Suite or try a different PC.
3. Make sure your firewall is turned off. More information on this can be found here: [Windows Firewall Configuration](#)
4. Make sure your roboRIO firmware is up to date using the instructions here: [Updating your roboRIO firmware](#)
5. Try a different PC
6. Try booting the roboRIO into Safe Mode by pressing and holding the reset button for at least 5 seconds.