

Moving from the Arduino Environment with Style

Presented by Doug Paradis

Copyright © Doug Paradis • www.dprg.org

This Workshop

- Two goals:
 - Help people move from the Arduino environment to Visual Studio Code / PlatformIO with GIT integration.
 - Show a programming style that uses some best practices and makes multifile projects easier.
- We have plenty of time. Everyone should achieve the goals and learn something.
- Just who is this "Doug Paradis" guy? I have been an active member of DPRG since 2010. I really enjoy designing and building robots. In another life, I was an Engineer at TI for 30 years.

Visual Studio Code by Microsoft

- Download link: <u>https://code.visualstudio.com/</u>
- We will look at two scenarios:
 - Using Visual Studio Code as an external editor for the Arduino Environment.
 - Using Visual Studio Code with PlatformIO to replace the Arduino Environment.
- Before we begin you should have the Arduino environment installed.

In the next section it is important that you use the right Arduino environment download



Change Arduino Sketchbook location for workshop

FileZil					x	
	File Edit Sketch Tools Help					
<u>(</u>		<u>2</u>			Ø	
eRout	test_external_edite	or				
Short	/* Blink				Î	
Prefere	ences					×
Settin	gs Network					_
Sketo	hbook location:					
D:\w	vin 10 \Documents \Arduino	_workspace			Browse	2
Edito	r language:	System Default v (requires restart of Arduino)				
Edito	r font size:	12				
Inter	face scale:	✓ Automatic 100 [∧] / _∨ % (requires restart of Arduino)				
Them	ne:	Default theme v (requires restart of Arduino)				
Show	verbose output during:	compilation 🗹 upload				
Comp	piler warnings:	None 🗸				
	Display line numbers	Enable Code Folding				
V	/erify code after upload	Use external editor				
v	Check for updates on star	tup Save when verifying or uploading				
	Jse accessibility features					
Addit	tional Boards Manager UR	Ls: vackage_pololu_index.json,http://dl.sipeed.com/MAIX/Maixduino/package_Maixduino_k210_index	ex.json			
More	preferences can be edite	ed directly in the file				
C:\Us	sers \parad \AppData \Loca	I\Arduino 15\preferences.txt				
(edit	only when Arduino is not	running)				
			C	к	Cance	el
	<pre>// the loop func void loop() {</pre>	tion runs over and over again forever			~	
Shorte	Invalid library f	ound in D:\winl0\Video\Video-2022\Workshop-Moving from Arduino env an	nd style-	-2022(0716\#	
						20
6	<	111			>	
	1		Arduino Uno	on CO	M6	
PeaZip	пошу не	Source Contractor			11-	

Visual Studio Code: Installing our first extension

• Step 1, install Arduino Extension by Microsoft



dep - 07/16/2022 Copyright © Doug Paradis • www.dprg.org

• Step 1 continued...



• Install set of helpful extensions





Will use later

• Step 2, open Arduino Sketchbook



• Navigate to Arduino Sketchbook



Upload • Attach Arduino board and select serial port Verifv Edit Selection View Go Run Terminal Help test external editor.ino - Arduino workspace - Visual Studio Code ... ⊕ 🖧 🐺 🖽 🖓 ₲ test external editor.ino × test external editor > 🚭 test external editor.ino > . PEN EDITORS 🗙 🚭 test external editor.ino test ex... by Colby Newman ~ arduino workspace 🗅 📴 ひ 🗗 This example code is in the public domain. > .vscode > libraries \sim test external editor > .vscode test external editor.ino // the setup function runs once when you press reset or power the board void setup() { // initialize digital pin LED BUILTIN as an output. Select file pinMode(LED_BUILTIN, OUTPUT); -0 // Setup Serial Serial.begin(115200); File ð // the loop function runs over and over again forever void loop() { Select Board type digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level) Serial.println("ON"); delay(1000); // wait for a second digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage Select Serial Port TERMINAL PROBLEMS OUTPUT DEBUG CONSOL Arduino **X** Maximum is 2048 bytes. IntelliSense configuration already up to date. To manually rebuild your IntelliSense ed figuration Access to run "Ctrl+Alt+I" > OUTLINE [Done] Analyzing sketch 'test_external_editor\test_external_edit_.ino' Examples > TIMELINE ARDUINO EXAMPLES ⊗0∆0 ⋒ Ln 9, Col 24 Spaces: 2 UTF-8 CRLF C++ <Select Programmer> test external editor\test external editor.ino Arduino Uno 🛱 COM6 Arduino 🔊 🗘

Open/Close Serial Monitor

11

dep - 07/16/2022 Copyright © Doug Paradis • www.dprg.org

• Open "blink"example



- Open "blink"example
- Add Serial Commands
 - Between lines 30-31 add: Serial.println("ON"):
 - Between lines 32-33 add: Serial.println("OFF");
- Open Serial Monitor
- Verify program
- Upload to Arduino Board
- Experiment with changing the program

Visual Studio Code: Quick tour of some features

- Open "blink"example
 - Auto Completion
 - IntelliSense
 - Go to Definition
 - Go to Declaration
 - Go to References
 - Format Document
 - Command Palette

• Using the "Command Palette" to change settings



• Using the "Command Palette" to change settings

Note:

This shows the Settings and Commands that can be made for the Arduino Extension.

The next Steps require that you attach an Arduino board to the computer.

S Fi	le Edit Selection View G	o Run Terminal Help Extension: Arduino - Visual Studio Code	
¢	exte 7 ° 🗟 ☴ …	≡ E >Arduino:	
	Arduino	Arduino: Open Serial Monitor recent	dy used 競
ر م	Arduino Arduino for Visual Stud… ∲Microsoft ऄऀ	Arduino: Cipicad Arduino: Library Manager Arduino: Verify Arduino: Celest Descuences	+ Alt + R
₽ a	Arduino ଦେ 83K Arduino support for Vi moozzyk Install	Arduino: Select Programmer Arduino: Select Serial Port Arduino: Change Board Type	vened
8	Arduino ଦ 54K VScode theme inspired lintangwisesa Install	Arduino CLI: Upload other Arduino CLI: Upload Using Programmer Arduino: Board Config	commands
-ø ₩	arduino-snipp � 79K Code snippets for ardu Ronaldo Sena Install	Arduino: Board Manager Arduino: Change Baud Rate	Categories
	Arduino Exte © 24K Collection of extension MPTY Install	 chat on gitter Welcome to the Visual Studio Code extension for Arduino preview ! The Arduino extension makes it easy to develop, build, deploy and debug your Arduino sketche in Visual Studio Code, with a rich set of functionalities. These include: IntelliSense and syntax highlighting for Arduino sketches Verify and upload your sketches in Visual Studio Code Built-in board and library manager 	Languages Debuggers Snippets
tho	Arduino_Theme		etches
	arduino-snipp Φ 4K A fork of arduino-snip Owen Flood Install AC Lab Arduino Φ 730		Extension Resources Marketplace Repository
8	Arduino development AC Lab Install	Built-in example list Built-in serial monitor	License Microsoft

• Using the "Command Palette" to change settings



• Using the "Command Palette" to change the Ardunio Serial Port

<	File Edit Selection View	Go Run Terminal Help Extension: Arduino - Visual Studio Code	
Ð	exte 🍸 Ü 🗮 …	≣ E >Arduino;	□ …
	Arduino	Arduino: Change Board Type recently used 😳	
		Arduino: Open Serial Monitor	
	Arduino 🕤 151ms	Arduino: Upload	
പ്പ	Arduino for visual Stud	Arduino: Library Manager	
		Arduino: Verify Ctrl + Alt + R	
	Arduino support for Vi	Arduino: Select Programmer	
<u>र</u> ू	moozzyk	Arduino: Select Serial Port	
	Arduino 00.54K	Arduino CLI: Upload other commands	-)enea.
	VScode theme inspired	Arduino CLI: Upload Using Programmer	15
	lintangwisesa Install	Arduino: Board Config	
L_⊗	arduino-snipp 🗘 79K	Arduino: Board Manager	Catalania
• •	Code snippets for ardu	Arduino: Change Baud Rate	Categories
•	Ronaldo Sena Install		Programming
	Arduino Exte 🗘 24K	chat on gitter	Languages
	Collection of extension		Debuggers Snippets
	MPTY Install	Welcome to the Visual Studio Code extension for Arduino ^{preview} I The Arduino	
Ê	Arduino_Theme 🛛 4K	extension makes it easy to develop build deploy and debug your Arduing settches	Formatters
.0	a color theme design b	in Visual Studio Code with a rich set of functionalities. These include:	
	kevinxing Install	in visual studio code, war a nell set of functionalities. These include.	Extension
	arduino-snipp 🗘 4K	 IntelliSense and syntax highlighting for Arduino sketches 	Resources
	A fork of arduino-snip	Verify and upload your sketches in Visual Studio Code	Marketplace
	owen Hood Install	Built-in board and library manager	Repository
	AC Lab Arduino \$\$730	Built-in example list	License
(8)	Arduino development	Built-in serial monitor	Microsoft

dep - 07/16/2022 Copyright © Doug Paradis • www.dprg.org

• Setting serial port





Git

- Free book "Pro Git" available on Amazon or at <u>https://git-scm.com/book/en/v2</u>.
- Great tutorial at https://backlog.com/git-tutorial/
- A Git repository tracks and saves the history of all changes (by snapshots) made to the files in a project.
- Very simple view of Git for a single contributor:
 - You create a repository (local) of your project in the project's folder (forms a ".git" folder). Commands: init and clone.
 - You make an initial commit of your project's files.
 - Select your files and "stage" them
 - Commit files to the repository and make a commit description (i.e., comment).
 - This forms a branch (master).
 - You make changes to your project files
 - Select the files that have changes (GIT extension options make this easy), stage them, and make a new commit and comment.
 - repeat...
- Works, but... It is easy to lose track of where you did what.



Git - Branches

- Branches can be used to help keep track of changes better and make working with partners easier. A branch is an independent line of development.
- Git workflows define a set of branch types that allow a project to advance and minimize issues with collaboration.
- The Git workflow we will use in the example has 5 types of branches: Master, Hotfix, Release, Develop, Features.

Master -- main branch that each commit is a project release

- **Hotfix** -- used when a quick fix in needed between releases
- **Release** -- created when about make a release, helps allow progress on the develop branches without interfering with the release
- **Develop** -- where all new features are collected before release
- Features -- type branch used for new features
- After this we will look at what would be typical for a normal hobby roboticist project.

Git - Branches

• Example robot project:

Branches are often named with their branch type as a prefix (ex., (Hotfix-Pen_selection_fix or H-Pen_Sel_fix)

B = branchM = MergeCheckout used tochange branch



Git - Branches

• Example robot project simplified for hobbyist single contributor:





Creating a web base copy of the repository

- It is very helpful to store the repository in the web.
 - It keeps the project safe in case of a problem with the local computer.
 - It allows up to date copies of the project on other computers.
 - It allows a central location for the project.
 - It allows collaborators to join the project easily.
- A site that you can store your repository is GitHub.
 - A repository stored on the web is called remote. The repository on your computer is called local.

• GitHub link: https://github.com/



• GitHub link: https://github.com/





dep - 07/16/2022 Copyright © Doug Paradis • www.dprg.org

• GitHub link: <u>https://github.com/</u>

Select the FREE option.



• GitHub link: https://github.com/



How to use Git video

• Very good beginning Git video: <u>https://www.youtube.com/watch?v=i_23KUAEtUM</u>

Watch video



• We have already installed the PlatformIO extension.



dep - 07/16/2022 Copyright © Doug Paradis • www.dprg.org

Click on



	File Edit Selection View Go Run Terminal	ielp platformio.ini - Blink_test - Visual Studio Code	– a ×
, i	EXPLORER ····	🤯 PIO Home 🛛 🏺 platformio.ini 🗙	□ …
	EXPLORER ···· V OPEN EDITORS Image: Constraint of the second seco	<pre> PIO Home</pre>	
Ę		t (Blink test)	ni 67 _O




Visual Studio Code with PlatformIO



Open Arduino environment

- Open Blink example
- Select all
- Copy
- Paste into main.cpp below #include <Arduino.h> line
- Move the Arduino.h statement to a more normal location below the initial comment.

Visual Studio Code with PlatformIO

≺	File Edit Selection View Go Run Terminal He	elp • main.cpp - Blink_test - Visual Studio Code	🔲 🗔 🛄 08 – 🗉 🗙
ſŊ	EXPLORER ····	🄯 PIO Home 🛛 🚭 main.cpp 8 🔍	₽≻ ↔ ∰ Ш …
	✓ OPEN EDITORS 1 unsaved	src > 😋 main.cpp >	
Q	🤯 PIO Home	9 If you want to know what pin the on-board LED is connected to	on your Arduin
	• 🕒 main.cpp src 8	10 model, check the Technical Specs of your board at:	
የዖ	✓ BLINK_TEST	11 https://www.arduino.cc/en/Main/Products	
8	> .pio	12	
	> .vscode	13 modified 2 Son 2016 by Antune Guadaluni	
₿Ż^	> include	15 modified 8 Sep 2016 by Colby Newman	
	> lib	16	
В	∽ src ●	17 This example code is in the public domain.	
	G• main.cpp 8	18	
	> test	19 https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink	
-0	♦ .gitignore	20 */	
3	🕺 platformio ini	21	
$\mathbf{\mathbf{v}}$	• •	22 #Include (Arduino.n)	
		24 // the setup function runs once when you press reset or power t	he board
		25 void setup() {	
		26 // initialize digital pin LED_BUILTIN as an output.	
		<pre>27 pinMode(LED_BUILTIN, OUTPUT);</pre>	
		28 }	
		29	
		30 // the loop function runs over and over again forever	
		32 digitalwrite(LED BUTLITN HIGH), // turn the LED on (HIGH i	s the voltage le
		33 delav(1000): // wait for a second	o the voltage it
Ø		34 digitalWrite(LED_BUILTIN, LOW); // turn the LED off by mak	ing the voltage
Ø		35 delay(1000); // wait for a second	
572	> OUTLINE	36 }	
స్టున			
*	×8∧0 ↔ ✓ → m ⊥ ΰ Σ 등 Default	(Blink test) In 22 Col 21 Spaces: 2 LITE-8 CRIE	C++ PlatformIO &

- Modify the delay statements
- Verify the code
- Upload the code

Did it work?

Visual Studio Code with PlatformIO – Using Libraries

We are not going to show how to add libraries to your project, however...

- PlatformIO has a mechanism for adding libraries that it has in its collection.
- Libraries can also be added by inserting them into the lib directory of the project. You would do this if the library is not in PlatformIO's library manager, or you have modified the library for some reason.
- The advantage of the built-in library mechanism of PlatformIO is that it will keep the libraries updated.
- A good link to learn about libraries in PlatformIO is: <u>https://www.youtube.com/watch?v=buFKeqbafDI</u>



dep - 07/16/2022 Cc

File Edit Selection View Go Run Terminal Help main.cpp - Blink_test - Visual Studio Code 🤯 PIO Home ♪~ ∰ ∏ … Germain.cpp X SOURCE CONTROL ſĽ src > G main.cpp > ... The folder currently open doesn't have a git If you want to know what pin the on-board LED is connected to on your Arduin Q repository. You can initialize a repository which model, check the Technical Specs of your board at: will enable source control features powered by https://www.arduino.cc/en/Main/Products مړ git. Make sure you modified 8 May 2014 by Scott Fitzgerald Initialize Repository saved file before modified 2 Sep 2016 by Arturo Guadalupi **a**/2 modified 8 Sep 2016 by Colby Newman To learn more about how to use git and source proceeding. control in VS Code read our docs. ₿ This example code is in the public domain. You can also directly publish this folder to a GitHub repository. Once published, you'll have <u>[</u>] access to source control features powered by git and GitHub. Ö #include <Arduino.h> 22 Publish to GitHub // the setup function runs once when you press reset or power the board void setup() { // initialize digital pin LED BUILTIN as an output. pinMode(LED BUILTIN, OUTPUT); // the loop function runs over and over again forever void loop() $+ \sim \wedge \times$ TERMINAL Build T... avrdude done. Thank you. $(\underline{8})$ > Upload... \ ર્જી * Terminal will be reused by tasks, press any key to close it.

×1 - I	File	Edit Selection View Go F	Run Terminal	Help	main.cpp - Blink_test - Visual Studio Code	🖵 🔲 08 – 💩 🗙
Ð		SOURCE CONTROL	Blink_test			£≻∽ ⇔ ⊞ …
_ م		The folder currently open doesn' repository. You can initialize a reț will enable source control features	Publish to Git Publish to Git powered by	tHub private tHub public r 10	repository () paradug/Blink_test epository () paradug/Blink_test model, cneck tne lecnnical specs of your board a Select pub	your Arduine
fo		git.		11 12	https://www.arduino.cc/en/Main/Products	
æ		Initialize Repository To learn more about how to use gi control in VS Code read our docs.	it and source	13 14 15 16	modified 8 May 2014 by Scott Fitzgerald modified 2 Sep 2016 by Arturo Guadalupi modified 8 Sep 2016 by Colby Newman	
		You can also directly publish this for GitHub repository. Once published	older to a d, you'll have	17 18 19	This example code is in the public domain. <u>https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink</u>	
- •		and GitHub.	bowered by git	20 21 22	*/ #include <arduino.h></arduino.h>	
۲		Ċ		23 24 25 26	<pre>// the setup function runs once when you press reset or power the void setup() { // initialize digital pin LED BUILTIN as an output.</pre>	poard
				27 28 29	<pre>pinMode(LED_BUILTIN, OUTPUT); }</pre>	
				30 31	// the loop function runs over and over again forever void loop() {	
				avrdud	L PROBLEMS OUTPUT DEBUG CONSOLE – e done. Thank you.	+ ~ ^ × • ▷ Build T ✓
				====== * Te	======================================	E Upload ✓
× (⊗ 0		🖞 🜔 🗟 Defa	ult (Blink_test)	Ln 22, Col 21 Spaces: 2 UTF-8 CRLF C	++ PlatformIO & Q

∢	File Edit Selection View Go Run Terminal H	lelp main.cpp - Blink_test - Visual Studio Code	08 – @ ×
Ch	SOURCE CONTROL ····	ioin pione Image: Comparison of the pione Image: Comparison of the pione Image: Comparison of the pione	~ 🗱 🔲 …
	imes source control	src > 😋 main.cpp >	
Q	Message (Ctrl+Enter to commit on 'master')	 9 If you want to know what pin the on-board LED is connected to on your Ar 10 model, check the Technical Specs of your board at: 	
مع	↔ Sync Changes	11 <u>https://www.arduino.cc/en/Main/Products</u>	
° ₽ ₽		12 13 modified 8 May 2014 by Scott Fitzgerald 14 modified 2 Sep 2016 by Arturo Guadalupi 15 modified 8 Sep 2016 by Colby Newman 16 17 This example code is in the public domain	
Ē		<pre>17 This example code is in the public domain. 18 19 https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink 20 */</pre>	
Ť		21 22 #include <arduino.h> 23</arduino.h>	
(b)		<pre>24 // the setup function runs once when you press reset or power the board 25 void setup() { 26 // initialize digital pin LED_BUILTIN as an output. 27 pinMode(LED_BUILTIN, OUTPUT); 28 }</pre>	
	> COMMITS > FILE HISTORY > BRANCHES	29 30 // the loop function runs over and code again forever 31 void loop() {	
8	STASHES STASHES	TERMINAL PROBLEMS OUTPUT DEP LG (Image: Second sec	to 檾 X
£633	Y TAGS WORKTREES SEARCH & COMPARE	Terminal will be reused by tas	en on GitHub

× F	ile Edit Selection View	Go Run Terminal H	elp • main.cp	p - Blink_test - Visual Studio Code			– a ×
C)	EXPLORER		♦ PIO Home src > main.cpp	Ioop() Ioop()		<mark>\$</mark> ≻~ ₿	; [] ···
	 PIO Home @ main.cpp src BLINK_TEST .pio .vscode include lib src @ main.cpp test .gitignore platformio.ini 	다 口 P	22 #includ 23 24 // the 25 void se 26 // in 27 pinMo 28 Seria 29 } 30 31 // the 32 void lo 33 digit 34 Seria 35 delay 36 digit 37 Seria 38 delay 39 }	<pre>de <arduino.h> setup function runs once etup() { hitialize digital pin LED_ bde(LED_BUILTIN, OUTPUT); el.begin(9600); loop function runs over a bop() { calWrite(LED_BUILTIN, HIGH el.println("ON"); /(1000); calWrite(LED_BUILTIN, LOW) el.println("OFF"); /(1000);</arduino.h></pre>	when you press reset or powe BUILTIN as an output. (); // turn the LED on (HIC // wait +- a sAdd s // wait +- a sAdd s // wait for a second	er the board 5H is the voltage l Serial statem making the voltage	
Shows repo	sitory		TERMINAL PROI	BLEMS OUTPUT DEBUG CONSO	LE GITLENS	+	~ ^ X
on master			avrdude done.	Thank you. ============== [SUCCESS] Took will be reused by tasks, p	3.33 seconds ====================================	• D Bui D Up	ild ⊺ ✓ oload ✓
*	> TIMEL NE I ^g master I ↔ ⊗ 0 ▲ 0 ⋒	✓ → m A ÿ	☑ ☑ Default (Bl	ink_test) Git Graph	Ln 37, Col 25 Spaces: 2 UTF-8	CRLF C++ PlatformIO	র্ম হ

dep - 07/16/2022 Copyright © Doug Paradis • www.dprg.org

How to change baud rate on serial monitor

⋞	File Edit Selection View Go Run	Terminal Help	 platformic 	o.ini - Blink_test - Visual Stu	idio Code		🔲 🖵 🛄 00 – 🗗 🗙
ſ	EXPLORER	··· 6+	main.cpp	ᅘ platformio.ini 🔍	🤯 PIO Home		↔ -○- ◇ (>) []] …
	 OPEN EDITORS 1 unsaved @ main.cpp src @ platformio.ini @ PIO Home BLINK_TEST .pio .vscode include lib src @ main.cpp test .gitignore @ platformio.ini 		<pre>platformio.ini You, 1 sec 1 ; Platf 2 ; 3 ; Bui 4 ; Upl 5 ; Lib 6 ; Adv 7 ; 8 ; Pleas 9 ; <u>https</u> 0 1 [env:un 2 platfor 3 board = 4 framewo 5 5 6 ; add t 7 monitor 8</pre>	cond ago 1 author (You) FormIO Project Conf Id options: build Load options: custo prary options: depe vanced options: ext se visit documentat c://docs.platformio m = atmelavr cuno prk = arduino co change baud rate c_speed = 115200	iguration File flags, source f m upload port, ndencies, extra ra script Are c ion for tra ct .org/p.ge/proje	ilter speed and extra library storage to the pla lebgen ball ctconf.html	flags es tformio.ini file clanate
(S) (S) (S) (S) (S) (S) (S) (S) (S) (S)	> OUTLINE > TIMELINE	Serial	Monit RMINAL PROP by the new cl D:\win10\D	D r, default b BLEMS OUTPUT DEBU ross-platform Power ocuments\PlatformIO	aud is 960 JG CONSOLE GITLE Shell https://a \Projects\Blink)() NS ka.ms/pscore6 _test>	≥ powershell + ∨ □ ₪ ^ ×
× .	\mathcal{V} master \bigcirc \bigotimes 0 \triangle 0 \bigcirc \checkmark \rightarrow	J A Q N	🔄 🗟 Default (Bl	ink_test) Git Graph	Ý You, 1 second	ago Ln 17, Col 23	Spaces: 4 UTF-8 CRLF Ini 🖗 🗘

⋞	File Edit Selection View Go Run Terminal	lelp • main.cpp - Blink_test - Visual Studio Code	🔲 🖵 🛄 08 – 🗉 ×
ſĴ	SOURCE CONTROL ····	🤯 PIO Home 🔮 main.cpp 🔍	<mark>⇔</mark> ≻~ ⇔ ⊞ …
	imes source control	src > 🚭 main.cpp > 😚 loop()	
P	Message (Ctrl+Enter to commit on 'master')	<pre>22 #include <arduino.h> 23</arduino.h></pre>	
የያ	igodot Sync Changes	24 // the setup function runs once when you press reset or power	the board
δ		25 Void setup() { 26 // initialize digital pin LED BUILTIN as an output.	
Ν		📑 🗟 📕 🔿 🗢 D:\win10\Documents\PlatformI0\Projects\Blink;test	X
æ⁄		File Hone Share View	^ @
-0	V COMMITS master	Pin to Quick Copy Paste Decision for the construction of the const	
Ш	91 Compare Working Tree with chranch tag	access resterior to	
	\bigcirc Up to date with origin on GitHub	← → ▼ ↑ → Libraries > Documents > Platform10 > Projects > Blink_test	✓ Č
Ľ.⊘́			rch vour
<u></u> 2	ک 🛔 (origin)> first commit You 6 minutes ago	Desktop * pio 7/15/2022.652 PM File folder	i cii youi
v		Documents a include 7/15/2022.652 PM File Folder project lo	cation, you will
		Decures # Inb //15/2022 6:52 PM File folder DPRG-RBIN-20220; src 7/15/2022 6:52 PM File folder	
		images lest 7/15/2022 6:52 PM File folder SEE UNAL II Slide decks igitignore 7/15/2022 6:52 PM Text Document 1 KB	. now nas a
		Workshop-Moving.	orv
		OneDrive - Personal	1
		This PC	
		Cubines 🚊 Documents	
	> FILE HISTORY	Music	
	> BRANCHES	₩ Videos	
	> REMOTES	Network	_
Q	> stashes	9 items	
	> TAGS	[SUCCESS] Took 2 23 seconds	> Upload ✓
563	> worktrees	* Terminal will be reused by tasks, press any key to close it.	
کیک	> SEARCH & COMPARE		
><	\mathcal{C} master $\overline{\mathcal{O}}$ \otimes 0 \triangle 0 $\widehat{\square}$ \checkmark \rightarrow $\widehat{\blacksquare}$ \blacksquare \bigcirc	Default (Blink_test) Git Graph Ln 37, Col 25 Spaces: 2 UTF-8 CR	_F C++ PlatformIO 🖗 🗘

ا 🖍	File Edit Selection View Go Run Terminal He	elp main.cpp - Blink_test - Visual Studio Code	🔲 🖵 🛄 08 – 🔿 🗙
Сŋ	SOURCE CONTROL ····	🄯 PIO Home 🕒 main.cpp M 🗙	\$ ^{>} ~ ∰ ∁ ↔ ↔ ⓑ Ш …
	 ✓ SOURCE CONTROL Message (Ctrl+Enter to commit on 'master') ✓ Commit ✓ Commit ✓ Changes ① ① main.cpp src M ✓ COMMITS master • Last fetched 1 minute ago ① Compare Working Tree with <branch, li="" t="" ①<=""> ② Compare Working Tree with <branch, li="" t="" ②<=""> ③ Up to date with origin on GitHub Last fetch > ④ (origin)> first commit You, 11 minutes ago </branch,></branch,>	<pre>src > C main.cpp > D loop() 22 #include <arduino.h> 23 24 // the setup function runs once when you press reset 25 void setup() { 26 // initialize digital pin LED_BUILTIN as an output. 27 pinMode(LED_BUILTIN, OUTPUT); 28 Serial.begin(9600); 29 } 30 31 // the loop function runs over and over again forever 32 void loop() { 33 digitalWrite(LED_BUILTIN, HIGH); // turn the LED 34 Serial.println("ON"); 35 delay(1000); // wait for a se 36 digitalWrite(LED_BUILTIN, LOW); // turn the LED 37 Serial.println("OFF"); You, now • Uncommitted 38 delay(1000); // wait for a se 39 } </arduino.h></pre>	or power the board
	> FILE HISTORY		
	> BRANCHES		
	> remotes	TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE GITLENS	+ ~ ^ ×
Q	> stashes	avrdude done. Thank you.	∎ Duild T ✓
	> TAGS	======================================	∑ Upload ✓
563	> WORKTREES	* Terminal will be reused by tasks, press any key to close	it
5	> SEARCH & COMPARE		
× 3	$ end{subarray}^{\circ} \oplus \otimes 0 \land 0 \land \checkmark \rightarrow \square \land \Downarrow \forall $	Default (Blink_test) Git Graph Ln 37, Col 25 Spaces: 2	UTF-8 CRLF C++ PlatformIO & 다

Save your changed file (or verify or run it) And you will see that a 1 in a blue circle shows up on the Git vertical tab.

>	-ile Edit Selection Viev	w Go Run Terminal He	lp main.cpp - Blink_test - Visual Studio Code		🔲 🖵 🛄 08 – 💩 🗙
Сŋ	SOURCE CONTROL		🄯 PIO Home 🚭 main.cpp M 🗙	\$~ €3	ኄ 🔶 🔶 🕀 🖽 …
	imes source control	王 🗸 羚 ひ 🖡 …	src > 🚭 main.cpp > 😚 loop()		
Q	Message (Ctrl+Enter to	commit on 'master')	22 #include <arduino.h> 23</arduino.h>		
<u>ړ</u>	√ Coi	mmit 🗸 🗸	<pre>24 // the setup function runs ond 25 void setup() {</pre>	e when you press reset or power t	the board
	∽ Changes	みりもり	26 // initialize digital pin LE	ED_BUILTIN as an output.	
å	G main.cpp src	rage All Cha	<pre>27 pinMode(LED_BUILTIN, OUTPUT) 28 Serial.begin(9600); 20 </pre>	;	
но Про	✓ COMMITS master • Last fet	tched 2 minutes ago	29 } 30		
	ርን Compare Working	」Tree with <branch, t="" th="" 以<=""><th>31 // the loop function runs over</th><th>r and over again forever</th><th></th></branch,>	31 // the loop function runs over	r and over again forever	
Ē	Up to date with or	rigin on GitHub Last fetch	32 void loop() { 33 digitalWrite(LED_BUILTIN, H]	IGH); // turn the LED on (HIGH :	is the voltage l ϵ
*	👌 🛕 🕻 origin 🗲 first c	ommit You, 12 minutes ago	34 Serial.println("ON");	// wait for a second	
			36 digitalWrite(LED BUILTIN, LC	DW); // turn the LED off by mal	king the voltage
			37 Serial.println("OFF");	You, now • Uncommitted changes	
A			38 delay(1000);	// wait for a second	
የኅ			39 }		
99					
●	> FILE HISTORY				
	> BRANCHES				±
	> remotes		TERMINAL PROBLEMS OUTPUT DEBUG CON	ISOLE GITLENS	
8	> stashes		avrdude done. Thank you.		D Build T ✓
	> TAGS		[SUCCESS] To	ook 3.33 seconds ====================================	[>] Upload √
563			* Terminal will be reused by tasks,	press any key to close it.	
×	$\begin{array}{c} \overline{} \\ \overline{} \\ \overline{} \\ \phantom{aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa$	命 ✓ → 歯 <u>A ♡</u>	S	Ln 37, Col 25 Spaces: 2 UTF-8 CRL	F C++ PlatformIO 🔗 🗘

Stage the changed file (main.cpp).

dep - 07/16/2022 Copyright © Doug Paradis • www.dprg.org

∢	File Edit Selection View Go Run Terminal	lelp main.cpp - Blink_test - Visual Studio Code	🔲 🖿 🛄 08 – 🙍 🗙
Ch	SOURCE CONTROL ····	Image: With the second seco	\$≻∨ ಱ಼ ဌ; ⊷ -⊳ ⇔ Ռ Ⅲ …
	SOURCE CONTROL Image: Control Image: Commit Changes Added serial messages Added serial messages Commit Image: Commit Changes Staged Changes Image: Commit Changes Staged Changes Image: Commit Changes Changes Image: Commit Changes	<pre> PIO Home</pre>	n (HIGH is the voltage le ond ff by making the voltage changes ond
19 ()	> FILE HISTORY	39 }	
	> BRANCHES	TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE GITLENS	+~ ^ ×
83 83 83	remotes stashes tags worktrees search a compare	avrdude done. Thank you. ====================================	■ Dild T ✓ Dild T ✓ Upload ✓ t.
×	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	부 🕞 🗟 Default (Blink_test) Git Graph 🛛 🛛 Ln 37, Col 25 Spaces: 2 U	TF-8 CRLF C++ PlatformIO & 다.

- Add commit message
- Commit to repository

∢	File Edit Selection View Go Run Terminal H	elp main.cpp - Blink_test - Visual Studio Code	🔲 🖵 🛄 08 – 🗉 ×
Ch	SOURCE CONTROL ····	😽 PIO Home 🕒 main.cpp X	\$\$; \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$ \$\$
	✓ SOURCE CONTROL	src > 🕒 main.cpp > 🛇 loop()	
Q	Message (Ctrl+Enter to commit on 'master')	22 #include <arduino.h> 23</arduino.h>	
وړ	$igodot$ Sync Changes 1 \uparrow	<pre>24 // the setup function runs once when you press reset or power 25 void setup() {</pre>	the board
å		<pre>26 // initialize digital pin LED_BUILTIN as an output. 27 pinMode(LED_BUILTIN, OUTPUT); 28 Serial.begin(9600); 29 }</pre>	
₽₽	✓ COMMITS 0↓ 1↑・master・Last fetche ら ひ 白 …	29 J 30	
	፟፟፟ጜ Compare Working Tree with <branch, t<="" td=""><td>31 // the loop function runs over and over again forever</td><td></td></branch,>	31 // the loop function runs over and over again forever	
Ē	> 🗇 Changes to push to ori 👘 n GitHub 1 co	32 void loop() {	is the veltage lo
ð	→ Added serial messages Yo Branch & master is > ↑ Added serial messages Yo Last fetched 5 minutes ago > \$ ▲	1↑ commit ahead of & origin/master on GitHub tes ago // wait for a second 36 digitalWrite(LED_BUILTIN, LOW); // turn the LED off by ma	aking the voltage
		37 Serial.println("OFF"); You, now • Uncommitted changes	
		38 delay(1000); // wait for a second	
ູໃງ		J	
(\mathbf{N})			
C	> FILE HISTORY		
	> BRANCHES	TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE GITLENS	+~ ^ ×
\sim	> REMOTES		
(8)	> STASHES		Dulla I V
~~~		======================================	
505	> SEARCH & COMPARE	Terminal will be reused by tasks, press any key to close it.	
×	$𝔅$ master ↔ 0↓ 1↑ ⊗ 0 $\triangle$ 0 $\triangle$ $\checkmark$ → 🛍 $△$	ヴ	F C++ PlatformIO 🔊 🗘

Note that head on local repository is one ahead compared to the repository on GitHub.

Sync Changes

≺	File Edit Selection View Go Run Terminal He	lp main.cpp - Blink_test - Visual Studio Code	🔲 🖵 🛄   00 – o x
Сh	SOURCE CONTROL ····	🤯 PIO Home 🛛 🚱 main.cpp	\$~~ \$ ↔ ↔ \$ □ …
_	imes source control	src > 📴 main.cpp > 🛇 loop()	
ρ	Message (Ctrl+Enter to commit on 'master')	22 #include <arduino.h> 23</arduino.h>	
مړ	igodot Sync Changes	24 // the setup function runs once when you press reset or point of setup() {	ower the board
		26 // initialize digital pin LED_BUILTIN as an output.	
$\leq_{\mathbf{f}}$		27 pinMode(LED_BUILTIN, OUTPUT);	
		28 Serial.begin(9600);	
₽₽	✓ <b>COMMITS</b> master • Last fetched 3 seconds ago	30	
LL	ር) Compare Working Tree with <branch, tag,<="" th=""><th>31 // the loop function runs over and over again forever</th><th></th></branch,>	31 // the loop function runs over and over again forever	
	🗅 Up to date with origin on GitHub Last fetch	32 void loop() {	
-9		33 digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (H	HIGH is the voltage l $\epsilon$
*	> 👲 ∢ origin >≻ Added serial messages You, 2	34 Serial.println("ON");	
	👌 👲 first commit You, 17 minutes ago	36 digitalWrite(LED_BUTLITIN, LOW): // turn the LED_off	hy making the voltage
		37 Serial.println("OFF"); You, now • Uncommitted chan	iges
		38 delay(1000); // wait for a second	
~		39 }	
Ϋ́			
	> FILE HISTORY		•
	> BRANCHES		$+ \times \wedge \times$
	> REMOTES		
$(\mathbf{A})$	> STASHES	avrdude done. Thank you.	
<u> </u>	> TAGS	======================================	≥ Upload ✓
563	> WORKTREES	* Terminal will be reused by tasks, press any key to close it.	
20	> SEARCH & COMPARE		
×	$\mathscr{P}$ master $\bigcirc$ $\otimes$ 0 $\triangle$ 0 $\bigcirc$ $\checkmark$ $\rightarrow$ $\square$ $\bigcirc$ $\bigcirc$	🕑 🗟 Default (Blink_test) Git Graph 🖕 You, now Ln 37, Col 25 Spaces: 2 UTF-8	CRLF C++ PlatformIO 🔗 🗘 🛓

#### After the sync the repository is the same locally and on GitHub

🔀 File Edit S	election View Go	Run Terminal Hel	p main.cpp	- Blink_test - Visu	al Studio Code			08 –	o ×
			🤯 PIO Home	🚱 main.cpp	×	$\sim \mathfrak{A}$	⊕ €	·O> (È)	□ …
<ul> <li>◇ OPEN EDI</li> <li>◇ OPEN EDI</li> <li>◇ P</li> <li>× G m</li> <li>&gt; .pio</li> <li>&gt; .yio</li> <li>&gt; .vscode</li> <li>&gt; include</li> <li>&gt; lib</li> <li>&gt; src</li> <li>G main.</li> <li>&gt; test</li> <li>&gt; .gitigne</li> <li>Ø platfor</li> <li>Ø platfor</li> </ul>	rors O Home ain.cpp src T cpp cpp		src > C main.cpp 22 #includ 23 24 // the 25 void se 26 // in 27 pinMo 28 Seria 29 } 30 31 // the 32 void lo 33 digit 34 Seria 35 delay 36 digit 37 Seria 38 delay 39 }	<pre>&gt; Doop() e <arduino.h (1000);="" (1000);<="" alwrite(led_="" de(led_built="" dig="" functi="" functio="" itialize="" l.begin(9600="" l.println("0="" loop="" op()="" pre="" setup="" tup()="" {=""></arduino.h></pre>	<pre>&gt; on runs once when you p ital pin LED_BUILTIN as IN, OUTPUT); ); n runs over and over ag BUILTIN, HIGH); // tu N"); // wa BUILTIN, LOW); // tu FF"); You, 7 minur // wa</pre>	press reset or power t s an output. gain forever urn the LED on (HIGH i ait for a second urn the LED off by mak tes ago • Uncommitted ait for a second	open Changes wi [Alt] Open Change ne board s the vol ing the v changes	th Previous Revision	n (Ctrl+Shift+G.)
S S S S OUTLINE > TIMELINE			TERMINAL PROB avrdude done. 	LEMS OUTPUT Thank you. ========================= /ill be reuse	DEBUG CONSOLE GITLENS SUCCESS] Took 3.33 seco d by tasks, press any k	nds ====================================		+ ~ ≥ Build ⊇ Uplo	X   T ✓ ad ✓
🚿 🧏 master 🕤		→ @ <u> </u>	🕑 🗟 Default (Bli	nk_test)  Git Gra	iph Ln 37, Col 2	25 Spaces: 2 UTF-8 CRLF	C++ Plat	tformIO	Å 🗘

Select the **C** Icon to compare the current commit to the Previous commit

⋈	File Edit	Selection	View Go	o Run	Terminal	Help	mai	n.cpp (ca7858	3) ↔ main.	cpp (8e423c9) - Blink	c_test - Visual Stu	idio Code			08 -	σx
Ð	🤯 Pio I	Home	😋 main.cp	р	🕒 main.e	cpp (ca78	583) ↔ main.c	рр (8е423с9)	க×			ţ; Ĉ [	£ ☆~ %	} ↑ י	↓ ¶ ′	↔ …
	D: > wi	n10 > Docun	nents > Pla	tformIO	> Projects 🕻	> Blink_tes	st > src > 📴 r	main.cpp								
ρ	26 27	// initi pinMode	ialize di (LED_BUIL	igital p TIN, OU	pin LED_E JTPUT);	BUILTIN	as an outp	ut.	26 27	<pre>// initializ pinMode(LED_</pre>	e digital p BUILTIN, OU	in LED_BUI TPUT);	LTIN as a	n outp	ut.	
የօ									$\rightarrow$ 28+	Serial.begin	(9600) <b>;</b>					
ۍ م	28 29	}							29 30	}						
	30	// the loo	op functi	ion runs	s over a	nd over	again fore	ver	31	// the loop fu	nction runs	over and	over agai	n fore	ver	
2	31	void loop	() {						32	<pre>void loop() {</pre>						
₿	32	digitalW	Nrite(LED	D_BUILTI	IN, HIGH	);	turn the L ////////////////////////////////////	ED on (HIC	33 → 34+	digitalWrite Serial.print	e(LED_BUILTI :ln("ON");	N, HIGH);	// turn	the L	ED on (	(HIG
	33	delay(10	300) <b>;</b>				wait for a	second	35	delay(1000);			// wait	for a	second	d
	34	digital	write(LED	D_BUILT	IN, LOW)	; []	turn the L	ED off by	36	digitalWrite	(LED_BUILTI	N, LOW);	// turn	the L	ED off	by
-0									$\rightarrow$ 37+	Serial.print	:ln("OFF");	You,	5 minutes	ago •		sei
3	35	delay(10	300) <b>;</b>			//	wait for a	second	38	delay(1000);			// wait	for a	second	a 🗖
	00	ſ							ور	1						
۲			<mark>Pre</mark>	vious	s com	mit					Current	comm	it			
06																
Υ.																
	TERMIN	AL PROBLI	ems out	IPUT D	EBUG CONS	ole git	TLENS								+ ~	^ X
$\bigcirc$	avrdu	de done.	Thank yo	u.										•	> Build	т 🗸
8															> Uploa	ad 🗸
~~~	=====			= [SUCC	ESS] Too	k 3.33 :	seconds ===		======	====						
રંદુર	* 1	erminal wi	III be re	used by	tasks,	press a	ny key to c	lose it.								
×	ှိ master	⊖ ⊗ <u>0⊿</u>	20 命	$\checkmark \rightarrow$	<u>گ</u>	Ϋ 🕥	🗟 Defaul <u>t (</u> Bl	link_test) Git	: Graph	∳ You,	5 minutes ago	Ln 37, Col 1	Spaces: 2	UTF-8	C++ ,	ନ 🗘

≺	File Edit Selection View Go	Run Terminal He	elp Git Graph - Blink_test - Visual Studio Code			08 🛄	- 0 X
Cŋ	EXPLORER						Ⅲ …
V OPEN EDITORS Git Graph: View Gi			Graph (git log) 🔫	recently used	C	<u>រភ</u>	- ጉ ስ
Q	🤯 PIO Home	GitLens: Show Bran	ch History		-		• •
	🕒 main.cpp src	PlatformIO: Serial N	Nonitor	Ctrl + Alt + S	Date	Author	Commit
وړ	🗙 👯 Git Graph	GitLens: Open Setti	ings		15 Jul 2022 2	paradug	8e423c9c
0	∨ BLINK_TEST	View: Show GitHub			15 Jul 2022 1	paradug	ca785835
Ν	> .pio	Arduino: Select Ser	ial Port			paradag	
æ^	> .vscode	Arduino: Select Ske	stch				
~0	> include	Arduino: Change B	oard Type				
Б	> lib	Arduino: Open Seri	al Monitor				
	∽ src	Arduino: Upload					
L_⊘	🕒 main.cpp	Arduino: Library Ma	anager				
	> test	Arduino: Verify		Altows you	to see		
Ö	♦ .gitignore			and a state of			
	🤯 platformio.ini			map of the	branches		
የኅ							
0 Ó							
			TERMINAL PROBLEMS OUTPUT DEBUG CONSOL	.e gitlens		+	~ ^ X
0			avrdude done. Thank you.			• 🖒 Bu	uild T 🗸
8						ν	pload 🗸
~~>			======================= [SUCCESS] Took	3.33 seconds ==========			
563			Terminal will be reused by tasks, pr	ess any key to crose it			
×	ழீ master 😌 🛞 0 🛆 0 🎧 🗸	→ 🛍 🛆 Ϋ	▷ ☐ Default (Blink_test) Git Graph				ጽ 🗘





Style – Everyone has one

- You should strive to develop your own "style" when programming unless you work for the man, then you do it his way.
- There are many guides to programming style on the web. You can learn a lot from them. Good style will help your program's readability. Remember you are doing this for fun and don't get caught up in the holy wars of different styles.
- There are two general styles of creating variable names: camelCase and snake_case. Examples: leftMotorPin versus left_motor_pin. You should try to be consistent.
- Constants should be all caps. Examples: int const LED = 13; or #define LED 13.
- You should develop a set of abbreviations that you use consistently program to program.
 Examples: mtr for motor, tm for time, flg for flag, ctr for counter, fwd for forward, rev for reverse, init for initiate, calc for calculate, meas for measure, etc ...

A Hobby Robot Multifile Project



Goal is to make 95% of the code reusable!

Making your programs consistent – use a template

- Since the files in our program often are being used in a similar way. It make sense that they have a lot of similarities in their structure.
- We can reinforce this by following a similar layout of the parts of the program in the file.
- For example, the order of parts of an .h file can be made so that each .h files look alike and are easy to write.
- We will look at main.cpp, .h files, bot_config.h specifically, and .cpp files.

Main.cpp template

```
The Arduino libraries:
SPI
PinChangeInterrupt
PID
```

This program also uses the following library: teensy encoder library - http://www.pjrc.com/teensy/td_libs_Encoder.html

```
Private program (not open source)
```

Revision: 20220612 Doug Paradis v0.0 - Start of project

Notes:

1. Heading can be determined either with IMU or by odometry using encoders.

Initial Comment

- Project name
- Description
- Expected Behavior
- Special Requirements
- Revisions (at least the starting date)
- Acknowledgements
- License / Copyright
- Notes

Main.cpp template

	<pre>#include <arduino.h> // required by PlatformIO</arduino.h></pre>	Top of
	<pre>#include <stdint.h></stdint.h></pre>	
	<pre>#include <math.h></math.h></pre>	
	<pre>#include "bot_config.h" #include "IMU_funcs.h"</pre>	• Inc • File
	<pre>#include "encoder_funcs.h"</pre>	
	<pre>#include "motor_funcs_VNH2HSP30_sign_mag.h"</pre>	
	<pre>#include "PID_loops.h"</pre>	
	<pre>#include "wpt_funcs.h"</pre>	
	<pre>#include "comm_funcs.h"</pre>	
	<pre>#include "nav_funcs.h"</pre>	
	<pre>#include "ultrasonic_funcs.h"</pre>	
	<pre>#include "helpful_defines.h"</pre>	
	// Global variables	
Use stdint	<pre>int16_t targ_edges_cnt [2] = {0,0};</pre>	
format	<pre>uint16 t loop_period = 20; // msec</pre>	
iuiiiau	<pre>uint32_t loop_cntr = 0;</pre>	omment units of
	Va	ariable

Top of file

- Include statements
- File globals

Main.cpp template

voi {	id setup()			
	// Setup Serial ports			
	Serial.begin(57600);	// wixel		
	Serial.println("Nav_test_for_clu	b_robot");		
	//note: Serial 1 pins are used a	s INT2 and INT3 for encouers	Always have main s	e
	<pre>Serial2.begin(115200);</pre>	// pixy	nort print pupique	
	<pre>Serial3.begin(57600);</pre>	// display	port print a unique	
			statement when	
	// initiate IMU			
	init_IMU(ptr_imu_sen);		initiated (lets you	
	// delay to allow IMU start up		Know what nom is	
	delay(3000);		Kilow what pgill is	
	// initiate Ultrasonic sensor		installed).	
	init_ultrasonic (ptr_us);		· · · · · · · · · · · · · · · · · · ·	
	<pre>// initiate motors</pre>			
	<pre>init_motors();</pre>			
	// initial trigger of US sensor			
	<pre>send_us_trig(ptr_us);</pre>			
	zero_encoders();	// set encoder cnts to	zero	
	<pre>open_gate(ptr_gate);</pre>	// start with open gate	e	

Setup()

- Setup serial ports
- Initiate all the parts of the robot
- Set starting settings

Loop()

• Run your loop

General .h file template

```
IMU funcs.h
   version: 20180504
                          Doug Paradis
   BN0055 functions
 ⊟#ifndef IMU FUNCS H
 #define IMU_FUNCS_H
 #include <Wire.h>
 #include <Adafruit_Sensor.h>
 #include <Adafruit BNO055.h>
 #include <utility/imumaths.h>
 // structs and enums
typedef struct IMU_sen_t
    uint8 t enable;
                                  // use to indicate IMU samp period reached
    uint8 t flg;
    uint16 t sample rate period;
                                  // mSec
    float yaw angle rad;
    float yaw_angle_deg;
    int16_t int_yaw_angle_deg;
    int16 t int x100 yaw angle deg;
    uint8_t connect_flg;
    Adafruit BN0055 sensor;
 }IMU_sensor;
 // externs
 extern IMU_sensor *ptr_imu_sen;
 //function prototypes
 void init_IMU (IMU_sensor *ptr_unit);
 void read IMU yaw(IMU sensor *ptr unit);
 #endif
```

- Initial comments
- #idndef,#define, #endif
- #includes
- Struts and enums
- Externs
- Function protypes for functions used by other parts of the program

General xx_funcs.cpp file template



- Initial comment
- #includes
- File scope globals
- Globals to be made into externs
- Local function prototypes
- Functions

bot_config.h template

<pre>/************************************</pre>
Author: Doug Paradis - 2/25/2020
Note: The Arduino PWM function of pins 10 and 9 can't be used with servo library

#ifndef BOT CONFIG H
#define BOT_CONFIG_H
<pre>#include <arduino.h></arduino.h></pre>
// constants
// Start hutton
const int16 t START BUTTON PIN = 53:
// Red and Green Turn LEDs
<pre>const uint8_t RED_LED_PIN = 49;</pre>
<pre>const uint8_t GREEN_LED_PIN = 48;</pre>
// Nation Controlling size - for size monitude (
// Motor Controller pins - for sign-magnitude < Club Robot
const int16 t PTN IN2 MTR = 9: // motor A input pin 2. PWM associated with
const int16 t PIN IN3 R MTR = 8; // motor B input pin 1
const int16 t PIN IN4 R MTR = 7: // motor B input pin 2. PWM associated wit

- Initial comment
- Set all pins names
- Set robot physical values (such as wheelbase)
- Set robot settings (such as speeds, slew)

bot_config.h template

```
// Pololu motors (enc cnts/rev = 3591.84)
                                                        // calc val = 0.00257
const float IN PER CLICK = 1 / L MTR EDGES PER IN;
/* Robot physical parameters:
   pulse/cm = (3591.11 edges/rev)Pololu Mtr / (31.416 cm/rev) = 114.3 edges/cm
   or 1/289.934 = 3.45 mils per pulse
const float WHEEL DIA MM = 96.5; // club robot // mm
const float WHEEL_DIA_CM = WHEEL_DIA_MM / 10.0;
const float WHEEL DIA IN = WHEEL DIA MM / 25.4;
const float ENC EDGES PER REV = 3591.11;
                                               // club robot - pololu mtrs (
const float R_MTR_EDGES_PER_CM = ENC_EDGES_PER_REV / (WHEEL_DIA_CM * PI); //
const float L MTR EDGES PER CM = ENC EDGES PER REV / (WHEEL DIA CM * PI);
const float R_MTR_EDGES_PER_IN = R_MTR_EDGES_PER_CM * 2.54;
const float L MTR EDGES PER IN = L MTR EDGES PER CM * 2.54;
// assumes that both clicks per in are the same and
const float WHEEL BASE IN = 9.4; //9.65; // Need to update if wheels
// Speeds, slews, etc...
const float OPEN LOOP TURN SPD = 60; // speed used when making final turn (ra
const float NORMAL_SPD = 15;
const float MAX TURN SPD = 20;
                                        // in/sec
const float TURN_SLEW = 1.0;
                                        // deg/sec
```

- Initial comment
- Set all pins names
- Set robot physical values (such as wheelbase)
- Set robot settings (such as speeds, slew)

Additional Rules

- No magic numbers
- No globals in functions
- No numbers representing true/false
- Every variable initiated to a value
- Use const instead of define
- All const/defines uppercase
- Always use #ifndef, #define, and #endif in .h files and around #defines
- Know the difference between a function passing by reference and passing by value
 - https://www.educative.io/answers/call-by-value-vs-call-by-reference-in-c
- Keep functions short and single purposed
- Make comments

Clone the repository that is at <u>https://github.com/paradug/Vendor_Provided_Robot_Code</u> Following example is from this program.

What is a magic number?



A magic number is a direct usage of a number in the code.

Problems

- Magic number (90)
- Function needs refactoring
- Global used in function

Note: This program has 3 similar functions (ask_pin_L, ask_pin_F, ask_pin_R) that should be made into one. We are keeping this function separate for teaching purposes.

Better...

```
float ask_pin_F(uint8_t servo_trig_pin, uint8_t servo_echo_pin) // Measure fwd distance
{
  float dist = 0;
   myservo.write(SERVO FWD VAL);
                                     // point ultrasonic sensor forward
   trig_servo(servo_trig_pin);
   float duration = pulseIn(servo echo pin, HIGH); // Read the time difference, usec
                                    // Speed of sound wave divided by 2 (go and back)
   dist = duration * SPD SOUND /2;
   Serial.print("F distance:");
                                    // Print servo measurement in cm
   Serial.println(dist);
   return(dist);
}
void trig_servo(uint8_t trig_pin)
{
   // Servo trigger requires a 10 ms (or greater) pulse
   digitalWrite(trig pin, LOW); // Low
   deLayMicroseconds(TRIG SETTLE TM);
   digitalWrite(trig_pin, HIGH); // Pulse
   deLayMicroseconds(TRIG PULSE LEN);
   digitalWrite(trig pin, LOW); // Low
}
```

• Note variable dist is initiated.

Or...

```
typedef struct us_sens_t {
    uint8_t enable;
    uint8_t trigger_pin_arduino;
    uint8_t echo_pin_arduino;
    uint8_t trig_settle_tm;
    uint8_t trig_pulse_tm;
    uint8_t triggered_flg;
    uint8_t new_read_flg;
    uint8_t unit_num;
    float dist_cm;
    float dist_in;
}us_sensor;
```

//externs

//extern us_sensor *ptr_us_R; //extern us_sensor *ptr_us_L; extern us_sensor *ptr_us;

- Using a typedef struct for representing an ultrasonic sensor.
- This would be in the .h file
- The .cpp file would have declared the variable
 *ptr_us and have init_us and read_us functions.

And...

```
void ask_pin_F(us_sensor ptr_sensor) // Measure the distance ahead
  uint16 t dist = 0;
  myservo.write(SERVO FWD VAL);
                                    // point ultrasonic sensor forward
  trig_servo(ptr_sensor->trigger_pin_arduino);
float duration = pulseIn(ptr sensor->echo pin arduino, HIGH); // usec
  ptr_sensor->dist_cm = duration * SPD_SOUND / 2; // Spd of sound divided by 2 (go and back)
  Serial.print("F distance:");
                                //The output distance (unit: cm)
  Serial.println(ptr sensor->dist cm);
                                              //According to the distance
}
void trig servo(us sensor ptr sensor)
  // Servo trigger requires a 10 ms (or greater) pulse
  digitalWrite(ptr_sensor->echo_pin_arduino, LOW);
  delayMicroseconds(2);
                                                        // settle time
  digitalWrite(ptr sensor->echo pin arduino, HIGH);
                                                        // pulse
  deLayMicroseconds(10);
                                                        // pulse len
  digitalWrite(ptr sensor->echo pin arduino, LOW);
                                                        // end pulse
}
```

- Using a typedef struct for representing an ultrasonic sensor.
- A course it would be better to only have one function that did the work of the three in the original code (ask_pin_F, ask_pin_R, and ask_pin_L).

Typedef structures that I typically use

- IMU_sensor *ptr_imu_sen;
- location *ptr_loc;
- camera *ptr_pixy;
- scoop_sensor *ptr_scoop_sen;
- us_sensor *ptr_us;
- temp_waypoint *ptr_temp_wpt;
- targets *ptr_targ;
- obstacle *ptr_obs;
- P4071_sensor *ptr_L4071;
Additional Rules – A last look

- No magic numbers
- No globals in functions
- No numbers representing true/false
- Every variable initiated to a value
- Use const instead of define
- All const/defines uppercase
- Always use #ifndef, #define, and #endif in .h files and around #defines
- Know the difference between a function passing by reference and passing by value

https://www.educative.io/answers/call-by-value-vs-call-by-reference-in-c

- Keep functions short and single purposed
- Make comments (put units in the comments)

