# アールティ製品共通 Arduino開発環境構築マニュアル

**1.2**版 株式会社アールティ



# 目次

目次	1
ご使用になる前に	2
コンピュータの動作要件	2
環境設定	3
コマンドの表記について	3
はじめに	3
Arduino IDEのインストール	3
Windowsの場合	6
Linux(Ubuntu)の場合	8
macOSの場合	10
Arduino IDEの初期設定	11
言語設定	11
外観設定	13
サンプルスケッチのビルドと書き込み	14
Pi:Co Classic4およびESP32-S3マイコンボード	14
ESP32ボード情報の追加	14
サンプルスケッチの準備	17
ビルド	21
書き込み	22
Pi:Co V2	25
ESP32ボード情報の追加	25
サンプルスケッチの準備	28
ビルド	32
書き込み	33
トレーニングトレーサー(TTV2)	36
STM32ボード情報の追加	36
書き込みツールSTM32CubeProgrammerのインストール	39
サンプルスケッチの準備	50
ビルド	54
書き込み	55
書き込み時のエラー対処方法	57
ST-Linkファームウェアのアップデート	58
使用しているツール、OSSのバージョン	66
改訂履歴	67
Copyright·知的財産権について	67



## ご使用になる前に

この度は、弊社の製品(以下「本製品」といいます)をお買い上げいただき、誠にありがとうございます。本製品をご使用になる前に、製品に付属する入門ガイドをお読みいただきますようお願いします。

## ■コンピュータの動作要件

本製品を使用するに当たり、Arduino IDEの動作要件を満たしたPCをご用意ください。Arduino IDEの動作要件は<u>https://www.arduino.cc/en/software</u>をご参照ください。



環境設定

ここでは本製品の開発環境の構築手順について説明します。

## コマンドの表記について

本書では、Linuxのターミナルで実行するコマンドを下記のように表記します。"\$"の後に続く文字列がコマンドを表します("\$"を除く)。"#"で始まる行はコメントを表すため実行不要です。

```
$ echo "コマンド"
# この行はコメントです
```

### はじめに

本製品はArduino Foundationが提供するArduino IDEを開発環境として使用します。Arduino IDEはマルチプラットフォームに対応しています。そのため、本製品は3種のOS(Windows、Linux 、macOS)で開発できます。

## Arduino IDEのインストール

2025年4月時点で最新版のArduino IDE 2.3.6をインストールする手順を示します。Arduino IDEが更新された場合は、本書内のバージョン番号を適宜読み替えてください。 webブラウザで<u>https://www.arduino.cc</u>にアクセスしProductsのArduino IDEをクリックします。

œ	For Professionals	For Education For Makers	1	cts ^ Community	✓ Documentation ✓ SHOP Q SHOP	Cloud
	HARDWARE BOARDS, SOMs and SBCS	KITS 😂 Starter Kit	SOFTWARE CLOUD AND TOOLS Arduino Claud		Arduino IDE Discover all the features of our most popular programming tool	>
	⇔ Giga ⇔ Nano ⇔ Portenta	<ul> <li>Plug &amp; Make Kit</li> <li>Science Kit</li> <li>Portenta Proto Kit NEN</li> </ul>	Arduine Claud Editor Arduine CLI Arduine IDE		Arduino Cloud Your next exciting journey to build and monitor your connected projects	>
	El Nicla ROBOTICS D: Alvik Robot	(5) PLC Starber Kit (5) Sensors Kit (5) Student Kit	PLC IDE APP IoT Remote		Cloud Editor Write code and upload sketches to any official Arduino board from your web browser	>
	INDUSTRIAL AUTOMATION	ALL HARDWARE $\rightarrow$	Science Journal ALL SOFTWARE →			

https://www.arduino.cc のトップページ

使用するOSに合うソフトウェアをDOWNLOAD OPTIONSからクリックします。本書では、 Windows MSI installer、Linux AppImage 64 bits(X86-64)およびmacOS Apple Silicon, 11: "Big Sur" or newer, 64 bitsを選択した前提でインストール手順を説明します。





以下のDownload Arduino IDE & support tis progressのページが表示された場合は、寄付をしてソフトウェアをダウンロードする"CONTRIBUTE & DOWNLOAD"と、ダウンロードのみを実施する"JUST DOWNLOAD"のどちらかを選択します。

PROFESSIONAL	EDUCATION	STORE			Q Search on A	rduino.cc		$\hat{\mathbb{O}}$	SIGN IN
©⊙	HARDWARE	SOFTWARE	CLOUD	DOCUMENTATION -	COMMUNITY -	BLOG	ABOUT		
	П	ownload	Ardui	no IDE & sun	port its pro	arece			
	Since t	he 1.x release	in March 2	015, the Arduino IDE h	as been downloade	d 77,959,	898		
		times –	– impressiv	ve! Help its developme	nt with a donation.				
		\$3 \$	\$5 \$	\$10 \$25	\$50 Other				
			CON	ITRIBUTE AND DOWN	LOAD				
				,001 201112012					
					C)				
			Learr	n more about <b>donating to An</b>	luino.				

Arduino IDE ダウンロード方法選択ページ



以下のStay in the loop: Join Our Newsletterのページでは、購読を希望する方はemailの記入 とチェックボックスの両方にチェックを入れ、SUBSCRIBE & DOWNLOADをクリックします。 購読 を希望しない方は、JUST DOWNLOADをクリックします。

s a b	eginner or advanced user, you can find inspiring projects and learn about cutting-edg Arduino products through our <b>weekly newsletter</b> !
	email *
	I confirm to have read the <b>Privacy Policy</b> and to accept the <b>Terms of</b> Service *
	l would like to receive emails about special deals and commercial offers from Arduino.
	SUBSCRIBE & DOWNLOAD
	or
	JUST DOWNLOAD



#### Windowsの場合

ダウンロードしたMSI installer(例: arduino-ide\_2.3.6\_Windows\_64bit\_msi)をダブルクリックしてインストールします。インストールが終わると下記の画面が消えます。

Arduino	IDE	
0	Please wait while Windows configures Arduino IDE	
		Cancel

インストール中の画面

Arduino IDEが正常にインストールできたことを確認するためArduino IDEを起動します。 Windowsのスタートボタン->検索ボックスに"arduino IDE"というキーワードを入力するとArduino IDEが検索結果として表示されます。

Q arduino ID	Ę					
< <b>ब</b> ∧र	עלת	۲ŧבאא	ウェブ	設定	フォルダー	写
最も一致する検索	結果					
💿 Arduino II アプリ	DE					
アプリ						

検索ボックスに"arduino IDE"を入力した状態



表示されたArduino IDEをクリックし、Arduino IDEが起動すると以下のような画面が表示されます。以上でインストール完了です。

🔤 ske	tch_mar5b	Arduino IDE 2.3.4	- 0 3	×
File Ec	dit Sketch	Tools Help		
Ø	€ 🔊	Select Board 👻	<u>م</u> ک	<b>D</b>
	sketch_m	ar5b.ino		
Ð	1 2 3 4	<pre>void setup() {     // put your setup code here, to run once:     } }</pre>		
	5 6 7	<pre>void loop() {     // put your main code here, to run repeatedly:</pre>		
0	8 9	}		l
Q	10			
8				
		Ln 1, Col 1 × No	board selected l	D

起動したArduino IDE

Windowsセキュリティの重要な警告メッセージが表示されたときは、アクセスを許可してください。アクセスをブロックしてしまうとArduino IDEのバージョンアップや追加ライブラリのダウンロードができなくなるためご注意ください。

ש Windows לאבין Windows ש	Jティの重要な警告		×				
このアプ ています	りの機能のいく け	こつかが Windows Defender ファイアウォールでブロックされ					
すべてのパブリックネ: の機能のいくつかがブ	ットワークとプライベー ロックされています。 名前(N):	ト ネットワークで、Windows Defender ファイアウォールにより Arduino IDE Arduino IDE					
	発行元(P):	Arduino SA					
	パス(H):	C:¥users¥maaok¥appdata¥local¥programs¥arduino-ide¥arduino ide.exe					
Arduino IDE にこれら	のネットワーク上で(	の通信を許可する:					
□ プライベート ネ	ットワーク (ホーム ネ	ットワークや社内ネットワークなど)(R)					
✓ パブリック ネッ (このようなネッ	✓ パブリック ネットワーク (空港、喫茶店など) (非推奨)(U) (このようなネットワークは多くの場合、セキュリティが低いかセキュリティが設定されていません)						
アプリにファイアウォールの経由を許可することの危険性の詳細							
		アクセスを許可する(A) キャンセル					
<b>k</b>	Windows-	セキュリティの重要な警告メッセージ	1				



#### Linux(Ubuntu)の場合

ダウンロードしたファイル(例:arduino-ide\_2.3.6\_Linux\_64bit.AppImage)を右クリックし、メ ニューからプロパティを選択します。プロパティ内のアクセス権タブを開き、"プログラムとして実行 可能"にチェックを入れます。

ダウン	Vロード ×	
Q		
	開く	Return
arduino-ide_2. 64bit.App	別のアプリケーションで開く( <u>A</u> )	
	切り取り( <u>T</u> )	Ctrl+X
	コピー( <u>c</u> )	Ctrl+C
	指定先へ移動	
	指定先にコピー	
	ゴミ箱へ移動する(⊻)	Delete
	名前を変更( <u>M</u> )	F2
	压縮( <u>O</u> )	
	星を付ける	
	プロパティ ( <u>R</u> )	Ctrl+I

AppImageファイルを右クリックした状態

arduino-ide_264bit.AppImage のプロパティ 🛛 🗙						
基本	アクセス権					
所有者(O)	自分					
アクセス	読み書き ~					
グループ(G)	ubuntu ~					
アクセス	読み書き ~					
その他						
アクセス	読み取り専用 〜					
実行	✔ プログラムとして実行可能(E)					
セキュリティコンテキスト	不明					

プロパティのアクセス権タブを開いた状態



その後、AppImageファイルをダブルクリックします。Terms of Serviceが表示された場合は、中身を確認し"Disagree/Agree"を選択してください。以上でインストール完了です。



Terms of Service画面

AppImageファイルをダブルクリックしてもArduino IDEが実行されない場合は、AppImageファ イルのコンテンツを展開し実行するツール(FUSE)がインストールされていない可能性がありま す。<u>https://github.com/AppImage/AppImageKit/wiki/FUSE</u>参照し、FUSEをインストールしてく ださい。



#### macOSの場合

ダウンロードしたファイル(例:arduino-ide\_2.3.6\_macOS\_arm64.dmg)をダブルクリックすると 以下の画面が表示されます。画面左にあるArduino IDEのアイコンを右のフォルダに移動すると インストールが完了します。インストール後のArduino IDEはLaunchpadから起動できます。



dmgファイルを開いた状態



### Arduino IDEの初期設定

ここではArduino IDEの言語や外観を設定します。また、本製品に搭載されているESP32-S3用のスケッチファイルをビルドするために、ライブラリやコンパイラを追加します。

#### 言語設定

Arduino IDEの言語を日本語に設定します。メニューバーにあるFile->Preferences...を選択します。



メニューバーのFile->Preferences...を選択した状態



Languageを"English"から"日本語"に変更し、右下にある "OK" をクリックします。

	Settings Network	
Sketchbook location:		
c:\Users\maaok\OneDrive\D	cuments/Arduino	BROWSE
Show files inside Sketche		
Editor font size:	14	
Interface scale:	Automatic 100 %	
Theme:	Light	
Language:	日本語 🗸 (Reload required)	
Show verbose output during	🗹 compile 🗌 upload	
Compiler warnings	None 👻	
Verify code after upload		
Auto save		
Editor Quick Suggestions		
Additional boards manager U	RLs:	6
		CANCEL

Languageを日本語に設定した状態

"OK"をクリックしたら自動的にArduino IDEが再起動します。再起動後、メニューバーの左端にあるボタンが"File"から"ファイル"になっていれば言語設定完了です。



#### 外観設定

OSの設定によっては、Arduino IDEの外観が暗い色調で表示されます。コントラストの弱いPC 画面では明るい色調に設定するとスケッチ編集画面のカーソルをはっきり表示できる場合があり ます。下記手順に従ってお好みの外観を設定してください。

メニューバーのファイル->基本設定…を選択します。"配色テーマ"を"Dark"から"Light"に変更 し、"OK"をクリックします。これによりArduino IDEの外観が明るい色調に変更されます。以降の ページで登場するArduino IDE画面は"Light"テーマを設定したものです。

基本設定			×
		設定 ネットワーク	
スケッチブックの場所:			
c:\Users\maaok\OneDrive\Docu	ments\Arduino		学版
□スケッチ内のファイルを表示	ŧ		_
エディターのフォントサイズ:	14		
インターフェイスのスケール:	☑ 自動 100	%	
配色テーマ:	Light	~	
エディター言語:	日本語	<ul> <li>(Reload required)</li> </ul>	
より詳細な情報を表示する	🛃 コンパイル(	□ 書き込み	
コンパイラの警告	なし *		
✓書き込み後にコードを検証す	3		
■ ロジネ(+)(0) □ エディターのクイックサジェ	スト		
 追加のポードマネージャのURL			
			-
			(キャンセル) OK(O)

配色テーマでLightを選択した状態



## サンプルスケッチのビルドと書き込み

ここでは各製品ごとにボードの追加、サンプルスケッチのダウンロード、ビルド、書き込み手順 を示します。

## Pi:Co Classic4およびESP32-S3マイコンボード



#### ESP32ボード情報の追加

スケッチを本製品に書き込めるように、ESP32のボード情報をArduino IDEに追加します。メ ニューバーのファイル->基本設定...を選択します。画面下部の"追加のボードマネージャのURL" に

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\_esp32\_ind ex.json

を追加し"OK"をクリックします。

		ſ	設定	ネットワーク
スケッチブックの場所:				
c:\Users\maaok\OneDrive\Doci	uments\Ardu	uino		(*) Ni
□スケッチ内のファイルを表示	R.			_
エディターのフォントサイズ:	14			
インターフェイスのスケール:	2日動	100	%	
配色テーマ:	Light		×	
エディター言語:	日本語		♥ (R	eload required)
より詳細な情報を表示する	2 コンパ	4110	書き込	74
コンパイラの警告	なし	~		
■書き込み後にコードを検証す	51			
✓ 自動保存(U)				
ロエディターのクイックサジュ	: 21-			
追加のボードマネージャのURL	https://rav	v.githubs	usercor	tent.com/espressif/arduino-esp32/gh-pages/package
				(taskil) ovio

追加ボードマネージャーのURLに追加した状態

URLを追加したら、ESP32ライブラリやツールをダウンロードしてインストールします。メニュー バーのツール->ボード->ボードマネージャ...を選択します。

🔤 ske	etch_mar58	o   Ardu	ino IDE 2.3.4			-			×
ファイル	編集 ス	ケッチ	ツール ヘルプ(H)						
	€		自動整形	Ctrl+T				$\checkmark$	·Q··
	sketch_	mar5	スケッチをアーカイブする ライブラリを管理	Ctrl+Shift+I					•••
白	2	VG	シリアルモニタ シリアルプロッタ	Ctrl+Shift+M		n once:			
lik	4 5	}	Firmware Updater SSLルート証明書を書き込み						
0	7		ボード		F	ボードマネージャ	Ctrl+	Shift+l	В
Q	9 10	}	ボート Reload Board Data ボード情報を取得		•	esp32 STM32 MCU based boards			•
			ブートローダを書き込む						

ボードマネージャ...を選択した状態



左側にあるボードマネージャの検索欄に"esp32"と入力すると"esp32 by Espressif Systems"が 表示されます。2025年4月時点では3.2.0が最新です。esp32 by Espressif Systemのインストー ルをクリックします。



ボードマネージャ画面の検索欄に"esp32"を入力した状態

Arduino IDEはESP32以外のマイコンにも対応しているため、スケッチを書き込む際はESP32 のボードを選択します。メニューバーのツール->ボード->esp32->ESP32S3 Dev Moduleを選択 します。"ボード"と"ポート"が同じような文字で上下にあるため間違えないように選択してください。

🔤 ski	etch_mar5b   Ard	luino IDE 2.3.4			-		×	ESP32P4 Dev Module
ファイル	編集 スケッチ	ツール ヘルプ(H)						ESP32H2 Dev Module
	<b>Ə</b> 🚯	自動整形	Ctrl+T			$\mathbf{v}$	ø	ESP32C6 Dev Module
	4. 11774	スケッチをアーカイブする						ESP32S3 Dev Module
	M-1-4-	ライブラリを管理	Ctrl+Shift+I					ESP32C3 Dev Module
_	esp32	シリアルモニタ	Ctrl+Shift+M	; your setup code	here,	to run	onc	ESP32S2 Dev Module
1	タイプ:全	シリアルプロッタ						ESP32 Dev Module
-	by Arduino	Firmware Updater						ESP32-WROOM-DA Module
ШИ	Boards inclu Arduino Nar	SSLルート証明書を書き込み		»() {			_	ESP32 Wrover Module
0	詳細情報	ボード		<ul> <li>ボードマネージャニ</li> </ul>	Ctrl	+ Shift + B	1	ESP32 PICO-D4
0	2.0.18- 🗸	ポート		esp32				ESP32S3 Dev Module Octal (WROOM2)
0	(1)7	Reload Board Data		STM32 MCU based box	ands			ESP32-S3-Box
Q		ポード情報を取得					-	ESP32-S3-USB-OTG
		ブートローダを書き込む						ESP32S3 CAM LCD
	esp32 by E	spressif					-	ESP32S2 Native USB

ツール->ボード->esp32->ESP32S3 Dev Modulesを選択

以上で、ESP32-S3のビルド環境設定完了です。



#### サンプルスケッチの準備

webブラウザで<u>https://github.com/rt-net/pico\_classic\_v4\_arduino\_examples</u>にアクセスし、本 製品のサンプルスケッチファイルをダウンロードします。

ページ内の"Code"から"Download ZIP"をクリックすることでダウンロードできます。ダウンロード したzipファイルは展開してください。

pico_classic_v4_arduino_examples	Public			🖈 Edit Pins 👻 💿	ch 2
🐉 main 👻 🎝 1 Branch 🛇 0 Tags		Q. Go to	file E	Add file ~ <> Cod	e -
🌑 YusukeKato CIの設定を追加 (#1) 🚥 🗸			Local	Codespaces	
github/workflows	CIの設定	を追加 (#1)			0
🖿 images					
pico_classic_v4_STEP1_LED	CIの設定を	を追加 (#1)	https://github.com/r	t-net/pico_classic_v4	Q
pico_classic_v4_STEP2_SWITCH	CIの設定を	を追加 (#1)	Clone using the web URL.		
pico_classic_v4_STEP3_Buzzer	CIの設定	を追加 (#1)	Gen with GitHub Des	sktop	
pico_classic_v4_STEP4_Sensor	2	C 1 1400	Download ZIP		

Code->Download ZIPを選択してサンプルスケッチをダウンロード

ここではPi Co Classic4のLEDを点灯させるサンプルスケッチ(pico\_classic\_v4\_STEP1\_LED) を例にして、スケッチのビルドと書き込み方法を説明します。

本製品にはUSB経由でスケッチを書き込みます。デフォルトではUSBからの読み書きができないため、読み書きができるように次の2箇所変更します。



まず、メニューバーのツール->ボード->esp32->ESP32S3 Dev Moduleを選択します。次に、メ ニューバーのツール->USB CDC On Bootを開き、"Enabled"にします。

ツール ヘルプ(H)		
自動整形	Ctrl+T	
スケッチをアーカイブする		
ライブラリを管理	Ctrl+Shift+I	_
シリアルモニタ	Ctrl+Shift+M	once:
シリアルプロッタ		
Firmware Updater		
SSLルート証明書を書き込み		
ボード: "ESP32S3 Dev Module"	)	<pre>epeatedly:</pre>
ポート	,	
ポード情報を取得		
USB CDC On Boot: "Disabled"	,	<ul> <li>Disabled</li> </ul>
CPU Frequency: "240MHz (WiFi)"	,	Enabled
Core Debug Level: "None"	,	

USB CDC On BootをEnabledに設定

メニューバーのツール->USB Modeを開き、"Hardware CDC and JTAG"にします。

F	ソール ヘルプ(H)		_	
Г	自動整形	Ctrl+T		
۰.	スケッチをアーカイブする			
	ライブラリを管理	Ctrl+Shift+I		
	シリアルモニタ	Ctrl+Shift+M	ance	
全	シリアルプロッタ			
	Firmware Updater			
0	SSLルート証明書を書き込み			
ch	ボード: "ESP32S3 Dev Module"		, epea	tedly:
lar	<i>#</i> - <b>ŀ</b>			
2	ポード情報を取得			
	USB CDC On Boot: "Enabled"		•	
	CPU Frequency: "240MHz (WiFi)"			
	Core Debug Level: "None"		•	
/ E	USB DFU On Boot: "Disabled"		•	
	Erase All Flash Before Sketch Upload: "Disabled"		•	
ch	Events Run On: "Core 1"		•	
CN	Flash Mode: "QIO 80MHz"		•	
ESI	Flash Size: "4MB (32Mb)"		•	
	JTAG Adapter: "Disabled"		•	
	Arduino Runs On: "Core 1"		•	
	USB Firmware MSC On Boot: "Disabled"		•	
	Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"		•	
	PSRAM: "Disabled"		•	
	Upload Mode: "UARTO / Hardware CDC"		•	
	Upload Speed: "921600"			
	USB Mode: "Hardware CDC and JTAG"		* ~	Hardware CDC and JTAG
	巻き込み装置		•	USB-OTG (TinyUSB)

USB ModeをHardware CDC and JTAGに設定



ダウンロードしたサンプルスケッチ内のpico\_classic\_v4\_STEP1\_LED/

pico\_classic\_v4\_STEP1\_LED.inoファイルをダブルクリックで開きます。または、Arduino IDEを 起動し、メニューバーのファイル->開く...をクリックしてpico\_classic\_v4\_STEP1\_LED.inoを開き ます。



ファイル->開く...を選択する

スケッチファイルを開く時に以下のメッセージが出た場合は、スケッチファイル名とフォルダ名が 一致していない、またはフォルダが無いということを示しています。OKをクリックして起動できる状態にします。

Arduino IDEの仕様上、メインとなる"ファイル名"と"フォルダ名"が同じではないときにこのメッ セージが表示されます。ファイル名やフォルダ名を変更する場合はそれぞれ同じ名前にしてくだ さい。



ファイル名とフォルダ名が一致しない場合の確認画面



サンプルスケッチを開くと以下のような画面が表示されます。

🔤 pic	o_classic_v4	_STEP1_LED   Arduino IDE 2.3.4	-		×
ファイル	編集スケ	ッチ ツール ヘルプ(H)			
	€	ボードを選択 ▼		≁	·Q··
Ph	pico_clas	ssic_v4_STEP1_LED.ino			
	1	// Copyright 2025 RT Corporation			
	2	//			_
t_)	3	<pre>// Licensed under the Apache License, Version 2.0 (the "License");</pre>			
	4	<pre>// you may not use this file except in compliance with the License.</pre>			
nh	5	<pre>// You may obtain a copy of the License at</pre>			
ШИ	6	//			
	7	<pre>// http://www.apache.org/licenses/LICENSE-2.0</pre>			
0	8				
	9	<pre>// Unless required by applicable law or agreed to in writing, software</pre>			
	10	<pre>// distributed under the License is distributed on an "AS IS" BASIS,</pre>			
Q	11	<pre>// WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.</pre>			
	12	<pre>// See the License for the specific language governing permissions and</pre>			
	13	// limitations under the License.			
	14				
	15	#define LEDG 12			
	10	#define LED0 15			
	19	#define LED2 47			
	10	#define LED3 48			
	20				
	21	void setup()			
	22	{			
	23	// put your setup code here, to run once:			
	24	<pre>pinMode(LED0, OUTPUT);</pre>			
	25	<pre>pinMode(LED1, OUTPUT);</pre>			
	26	<pre>pinMode(LED2, OUTPUT);</pre>			
	27	<pre>pinMode(LED3, OUTPUT);</pre>			
	28	}			
	29				
Q	30	void loop()			
-	31	{			
		行1、列1 >	× ボード未	· 違択 C	

pico\_classic\_v4\_STEP1\_LED.inoを開いた状態

ボードが"ESP32S3 Dev Module"になっていないときは、メニューバーのツール->ボード ->esp32->ESP32S3 Dev Moduleを選択してください。

🔤 ske	rtch_mar5b   Ard	uino IDE 2.3.4			-		×	ESP32P4 Dev Module
ファイル	編集 スケッチ	ツール ヘルプ(H)						ESP32H2 Dev Module
	<b>e</b> 🔊	自動整形	Ctrl+T			$\mathbf{v}$	ø	ESP32C6 Dev Module
	4. 1774	スケッチをアーカイブする						ESP32S3 Dev Module
	- r <*-	ライブラリを管理	Ctrl+Shift+I		0.1			ESP32C3 Dev Module
	esp32	シリアルモニタ	Ctrl+Shift+M		:up() { : your setup code here, 1	to rur	n onc	ESP32S2 Dev Module
1	タイプ: 全	シリアルブロッタ						ESP32 Dev Module
	by Arduino	Firmware Updater						ESP32-WROOM-DA Module
	Boards inclu Arduino Nar	SSLルート証明書を書き込み			æ() {			ESP32 Wrover Module
	詳續情報	ボード		•	ポードマネージャ Ctrl	+Shift+	В	ESP32 PICO-D4
0	2.0.18- 🛩	ポート		۲	esp32			ESP32S3 Dev Module Octal (WROOM2)
~	127	Reload Board Data			STM32 MCU based boards			ESP32-S3-Box
Q		ポード情報を取得					-	ESP32-S3-USB-OTG
		ブートローダを書き込む						ESP32S3 CAM LCD
	esp32 by E	spressif					_	ESP32S2 Native USB

ツール->ボード->esp32->ESP32S3 Dev Moduleを選択



ビルド

左上にあるアイコンで"検証"を実行しスケッチをビルドします。スケッチの記述に問題がなければ以下のようなメッセージが表示されます。

出力

最大1310720バイトのフラッシュメモリのうち、スケッチが248565バイト(18%)を使っています。 最大327680バイトのRAMのうち、グローバル変数が18456バイト(5%)を使っていて、ローカル変数で309224バイト使うことができます。

```
検証に問題が無いときのメッセージ
```

スケッチの記述にエラーがあった場合は、赤文字でエラー箇所が表示されます。

出力
"C:\\Users\\maaok\\AppData\\Local\\Arduino15\\packages\\esp32\\tools\\esp-x32\\2405/bin/xtensa C:\Users\maaok\Downloads\pico_classic4_arduino_examples-main\pico_classic4_arduino_examples-ma C:\Users\maaok\Downloads\pico_classic4_arduino_examples-main\pico_classic4_arduino_examples-main
27   pinMode(LED4, OUTPUT);   ^~~~   LED0
exit status 1
Compilation error: 'LED4' was not declared in this scope; did you mean 'LED0'?

検証エラー時のメッセージ

Arduino IDEの右下にあるアイコン 一度クリックすると表示されます。画面を大きくしてスケッチ全体を少しでも多く見たい時や、この メッセージ画面が不要なとき等に活用してください。

Linux(Ubuntu)の場合、ModuleNotFoundError: No module named 'serial'のメッセージが出 カされたときはビルドができていません。下記のコマンドをターミナルで実行し、ビルドに必要なア プリをインストールしてください。

\$ sudo apt-get install python-pip \$ pip install pyserial \$ pip install esptool



#### 書き込み

スケッチにエラーが無いことを確認したら、次の手順で書き込み前の準備をします。

- 1. 本製品の電源スイッチをOFFにします。
- 2. 本製品にバッテリを接続します。
- 3. 本製品の書き込み端子とPCをUSBケーブルで接続します。
- 4. 本製品のモード切替スイッチをRUN1とは逆側の方向(下図の赤い印の方)にスライドします。
- 5. 本製品の電源スイッチをONにします。
- 6. 本製品のリセットボタンを押します。



モード切替スイッチを操作し、USBケーブルを接続した状態



本製品をPCと接続した後、以下の手順でボードとポートを設定します。本製品の電源をOFFに した後や、リセットボタンを押した後にも再度設定してください。

メニューバーの下に表示されているボード名(ESP32S3 Dev Module)をクリックし、一番下の" 他のボードとポートを選択…"を開きます。



ボード名->他のボードとポートを選択...を開く

ボードの検索欄に"dev"を入力し、"ESP32S3 Dev Module"にチェックが入っていることを確認し ます。チェックが入っていない場合は、ESP32S3 Dev Moduleをクリックします。ポート選択欄で は本製品を接続したUSBポートを選択します。Windowsの場合はCOMX、Linux(Ubuntu)の場 合は/dev/ttyUSBX、macOSの場合はusbmodemXXXXX Serial Port(USB)という名称です。選 択したら"OK"をクリックします。

他のボードとポートを選択		×
スケッチを書き込みたい場合には、ボードとれ ボードのみを選択した場合、コンパイルはでき	ポートのi きますが、	両方を選択してください。 、スケッチの書き込みはできません。
ボード		ポート
dev	Q	
ESP32H2 Dev Module		COM4 Serial Port (USB)
ESP32S2 Dev Module	- 1	COM1 Serial Port
ESP32S3 Dev Module	~	
ESP32S3 Dev Module Octal (WROOM2)		
Hornbill ESP32 Dev		
Lion:Bit Dev Board		
		(キャンセル) OK(0)

ボードとポートの設定



 $\checkmark$   $\rightarrow$  (設定完了後、ボート名の左にUSBのマーク が追加 されます。スケッチの書き込み以外に、PCとハードウェアシリアル通信をするときも、ポート設定 が必要です。



ポート設定後、 をクリックしてスケッチを書き込みます。下記のように表示されたら書き込 み完了です。



書き込み完了時の画面表示

Linux(Ubuntu)PCで書き込みに失敗する時は、書き込み前にターミナルで次のコマンドを実行 し、USBポートの使用権限を変更します。

\$ sudo chmod 777 /dev/ttyACM0

永続的にUSBポートの使用権限を変更する場合は、次のコマンドを実行し、PCを再起動しま す。

```
# 下記を実行した後にPCを再起動すること
```

\$ sudo usermod -aG dialout \$USER

スケッチの書き込み完了後、本製品のモード切替スイッチをRUN1側にスライドし、リセットボタ ンを押します。

本製品の前方のLEDが0.5秒点滅、0.5秒消灯を繰り返します。





## ESP32ボード情報の追加

スケッチを本製品に書き込めるように、ESP32のボード情報をArduino IDEに追加します。メ ニューバーのファイル->基本設定…を選択します。画面下部の"追加のボードマネージャのURL" に

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package\_esp32\_ind ex.json

を追加し"OK"をクリックします。

基本設定	3	×
	設定 ネットワーク	
スケッチブックの場所:		
c:\Users\maaok\OneDrive\Docu	iments/Arduino 👘	
□スケッチ内のファイルを表示		
エディターのフォントサイズ:	14	
インターフェイスのスケール:	2 自動 100 %	
配色テーマ:	Light v	
エディター言語:	日本語 V (Reload required)	
より詳細な情報を表示する	図 コンパイル □ 書き込み	
コンバイラの警告	なし マ	
☑ 書き込み後にコードを検証す	8	
✓ 自動保存(U)		
」エティターのクイックサジェ きれのポードフラーミックリアレ	X N https://www.aithubusassantast.com/assess/i/arduine.com/20/ah.com/ass/aas/aas/	
道加のハートマネーシャのURL	nttps.//raw.githubusercontent.com/espressit/arouno-esp32/gn-pages/package	
	(キャンセル) OK(O)	b.
		ſ.,

追加ボードマネージャーのURLに追加した状態



URLを追加したら、ESP32ライブラリやツールをダウンロードしてインストールします。メニュー バーのツール->ボード->ボードマネージャ...を選択します。

🔤 ske	tch_mai	r5b /	Ardu	ino IDE 2.3.4				- 0	×
ファイル	編集	スケッ	Ŧ	ツール ヘルプ(H)					- L.
Ø	€ (		Γ	自動整形	Ctrl+T			$\checkmark$	۰ <b>©</b> ۰۰
	sketc 1 2 3 4	h_ma L 2 3 4	ar5 vc }	スケッチをアーカイブする ライブラリを管理 シリアルモニタ シリアルプロッタ Firmware Updater	Ctrl+Shift+I Ctrl+Shift+M		n once:		
0	2 7 8 9 10	> 7 3 9	}	ボード ポート Reload Board Data ポード情報を取得 ブートローダを書き込む		•	ボードマネージャ esp32 STM32 MCU based board	Ctrl+Shift+	B •

ボードマネージャ...を選択した状態

左側にあるボードマネージャの検索欄に"esp32"と入力すると"esp32 by Espressif Systems"が 表示されます。2025年4月時点では3.2.0が最新です。esp32 by Espressif Systemのインストー ルをクリックします。



ボードマネージャ画面の検索欄に"esp32"を入力した状態



Arduino IDEはESP32以外のマイコンにも対応しているため、スケッチを書き込む際はESP32 のボードを選択します。メニューバーのツール->ボード->esp32->ESP32S3 Dev Moduleを選択 します。"ボード"と"ポート"が同じような文字で上下にあるため間違えないように選択してください。

🔤 ske	etch_jun3b   Ardui	ino IDE 2.3.2			ESP32H2 Dev Module
ファイル	編集 スケッチ	ツール ヘルプ(H)			ESP32C6 Dev Module
	<b>Ə</b> 🚯 [	自動整形	Ctrl+T		ESP32S3 Dev Module
	#_ 572-	スケッチをアーカイブする			ESP32C3 Dev Module
		ライブラリを管理	Ctrl+Shift+I	un() I	ESP32S2 Dev Module
_	esp34	シリアルモニタ	Ctrl+Shift+M	; your setup code here, to	ESP32 Dev Module
白	タイプ: 全	シリアルプロッタ			ESP32-WROOM-DA Module
-	Andulas F	Firmware Updater			ESP32 Wrover Module
UNA.	by Arduino	SSLルート証明書を書き込み		»p() {	ESP32 PICO-D4
~	Boards inclu	ボード	,	ポードマネージャ Ctrl+Shift+B	ESP32S3 Dev Module Octal (WROOM2)
0	Arduino Nar 詳細情報	ポート	,	esp32 +	ESP32-S3-Box
0	2013 ¥	ポード情報を取得		-	ESP32-S3-USB-OTG
Q	177	ブートローダを書き込む			ESP32S3 CAM LCD
					ESP32S2 Native USB

ツール->ボード->esp32->ESP32S3 Dev Modulesを選択

以上で、ESP32-S3のビルド環境設定完了です。



#### サンプルスケッチの準備

webブラウザで<u>https://github.com/rt-net/pico\_v2\_arduino\_examples</u>にアクセスし、Pi:Co V2 のサンプルスケッチファイルをダウンロードします。

ページ内の"Code"から"Download ZIP"をクリックすることでダウンロードできます。ダウンロード したzipファイルは展開してください。

rt-net / pico_v2_arduino_ex	xamples Public	🗘 Notifi
<> Code 💿 Issues 🕕 🛱 Pull re	equests 💿 Actions 🖽 Projects 🙂 Security 🗠 Insig	
양 main ▾ 양 1 Branch ⓒ 0 Tags	Q Go to file	Code -
YusukeKato READMEICarduino-esp	p32のバージョ	7
github/workflows	micrc GitHub CLI	
pico_v2_STEP1_LED	サン: https://github.com/rt-net/pico_v2_arduino_exam	ч С
pico_v2_STEP2_SWITCH	Clone using the web URL サン:	
pico_v2_STEP3_Buzzer	サン: 🔄 Open with GitHub Desktop	
pico_v2_STEP4_Sensor (2)	Download ZIP	
pico_v2_STEP5_Straight	micro step 1/4 -> 1/8 change (#4) 2 0	days ago

Code->Download ZIPを選択してサンプルスケッチをダウンロード

ここではPi:Co V2のLEDを点灯させるサンプルスケッチ(pico\_v2\_STEP1\_LED)を例にとって スケッチのビルドと書き込み方法を説明します。

本製品にはUSB経由でスケッチを書き込みます。デフォルトではUSBからの書き込みができないため、書き込みができるように次の2箇所変更します。

まず、メニューバーのツール->ボード->esp32->ESP32S3 Dev Moduleを選択します。次に、メ ニューバーのツール->USB CDC On Bootを開き、"Enabled"にします。



ツール ヘルプ(H)	
自動整形	Ctrl+T
スケッチをアーカイブする	
ライブラリを管理	Ctrl+Shift+I
シリアルモニタ	Ctrl+Shift+M once:
シリアルプロッタ	
Firmware Updater	
SSLルート証明書を書き込み	
ポード: "ESP32S3 Dev Module"	<pre>▶ epeatedly:</pre>
ポート	• •
ポード情報を取得	
USB CDC On Boot: "Disabled"	<ul> <li>Disabled</li> </ul>
CPU Frequency: "240MHz (WiFi)"	Enabled
Core Debug Level: "None"	
USB CDC On Boot	をEnabledに設定

メニューバーのツール->USB Modeを開き、"Hardware CDC and JTAG"は	します。
--	------

۶.	ゾール ヘルプ(H)		
Г	自動整形	Ctrl+T	
	スケッチをアーカイブする		
	ライブラリを管理	Ctrl+Shift+I	
	シリアルモニタ	Ctrl+Shift+M	ance:
全	シリアルプロッタ		
	Firmware Updater		
0	SSLルート証明書を書き込み		
clu	ボード: "ESP32S3 Dev Module"	•	<pre>&gt;peatedly:</pre>
Lar.	<i>#</i> -ŀ	,	
	ポード情報を取得		
2	USB CDC On Boot: "Enabled"	•	
	CPU Frequency: "240MHz (WiFi)"	,	
	Core Debug Level: "None"	,	
/ E	USB DFU On Boot: "Disabled"	,	
	Erase All Flash Before Sketch Upload: "Disabled"	•	
c la	Events Run On: "Core 1"	•	
CV.	Flash Mode: "QIO 80MHz"	•	
esi	Flash Size: "4MB (32Mb)"	•	
	JTAG Adapter: "Disabled"	•	
-	Arduino Runs On: "Core 1"	•	
	USB Firmware MSC On Boot: "Disabled"	•	
	Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"	•	
	PSRAM: "Disabled"	•	
	Upload Mode: "UARTO / Hardware CDC"	•	
	Upload Speed: "921600"		
	US8 Mode: "Hardware CDC and JTAG"	•	✓ Hardware CDC and JTAG
	巻き込み検査	,	USB-OTG (TinyUSB)

USB ModeをHardware CDC and JTAGに設定

ダウンロードしたサンプルスケッチ内のpico\_v2\_STEP1\_LED/pico\_v2\_STEP1\_LED.inoファイ ルをダブルクリックで開きます。または、Arduino IDEを起動し、メニューバーのファイル->開く…を クリックしてpico\_v2\_STEP1\_LED.inoを開きます。

🥯 sketch_marба   Arduino IDE 2.3.4									
ファイル 編集 スケッチ	ツ−ル ヘルプ(H)								
新規スケッチ	Ctrl+N								
新規クラウドスケッチ	Alt+Ctrl+N								
開く	Ctrl+O								
最近使った項目を開く	•	etup code her							
スケッチブック	•								
スケッチ例	•								
閉じる	Ctrl+W								
Save	Ctrl+S	ain code here							

ファイル->開く...を選択する

スケッチファイルを開く時に以下のメッセージが出た場合は、スケッチファイル名とフォルダ名が 一致していない、またはフォルダが無いということを示しています。OKをクリックして起動できる状態にします。

Arduino IDEの仕様上、メインとなる"ファイル名"と"フォルダ名"が同じでないときにこのメッセージが表示されます。ファイル名やフォルダ名を変更する場合はそれぞれ同じ名前にしてください。

移動	×	
i	ファイル"pico_v2_STEP1_LED.ino"は、"pico_v2_STEP1_LED"という名前のスケッチフォルダの中にあるこ とが必要です。 このフォルダを作成し、ファイルを移動させ、継続しますか?	
	→ キャンセル	
	$\rightarrow$ OK(O)	
<b>b</b>	ファイル名とフォルダ名が一致しない場合の確認画面	



サンプルスケッチを開くと以下のような画面が表示されます。

🔤 pice	pico_v2_STEP1_LED   Arduino IDE 2.3.2						
ファイル	編集 スク	ッチ ツール ヘルプ(H)					
$\bigcirc$	€	ポードを選択 ▼		$\checkmark$	۰ <b>©</b> ۰۰		
P	pico_v2	_STEP1_LED.ino					
	1	// Copyright 2023 RT Corporation					
-	2						
1-)	3	<pre>// Licensed under the Apache License, Version 2.0 (the "License");</pre>					
	4	// you may not use this file except in compliance with the License.					
Ith	5	// You may obtain a copy of the License at					
UTIO	6						
	7	// http://www.apache.org/licenses/LICENSE-2.0					
0	8	//					
	9	// Unless required by applicable law or agreed to in writing, software					
0	10	// WITHOUT WARPANTIES OF CONDITIONS OF ANY KIND withon express on implied					
X	12	// See the license for the specific language governing permissions and					
	13	// limitations under the License.					
	14	,					
	15	#define LED0 42					
	16	#define LED1 41					
	17	#define LED2 15					
	18	#define LED3 16					
	19						
	20	void setup()					
	21	{					
	22	<pre>// put your setup code here, to run once:</pre>					
	23	<pre>pinMode(LED0, OUTPUT);</pre>					
	24	pinMode(LED1, OUTPUT);					
	25	pinMode(LED2, OUTPUT);					
	20	pinMode(LED3, OUTPOI);					
	27	3					
	29	void loop()					
	30	{					
	31	// put your main code here, to run repeatedly:					
	32	<pre>digitalWrite(LED0, HIGH);</pre>					
	33	<pre>digitalWrite(LED1, HIGH);</pre>					
	34	<pre>digitalWrite(LED2, HIGH);</pre>					
	35	<pre>digitalWrite(LED3, HIGH);</pre>					
	36	delay(500);					
	37	<pre>digitalWrite(LED0, LOW);</pre>					
	38	digitalWrite(LED1, LOW);					
	39	digitalWrite(LED2, LOW);					
	40	digitalWrite(LED3, LOW);					
	41	1					
	43						
	45						
Q							
0							
		行1.列	1 × ボー	下未译排	20		

pico\_v2\_STEP1\_LED.inoを開いた状態

ボードが"ESP32S3 Dev Module"になっていないときは、メニューバーのツール->ボード ->esp32->ESP32S3 Dev Moduleを選択してください。



ツール->ボード->esp32->ESP32S3 Dev Moduleを選択



ビルド

左上にあるアイコンで"検証"を実行しスケッチをビルドします。スケッチの記述に問題がなければ以下のようなメッセージが表示されます。

出力

最大1310720バイトのフラッシュメモリのうち、スケッチが248565バイト(18%)を使っています。 最大327680バイトのRAMのうち、グローバル変数が18456バイト(5%)を使っていて、ローカル変数で309224バイト使うことができます。

検証に問題が無いときのメッセージ

スケッチの記述にエラーがあった場合は、赤文字でエラー箇所が表示されます。



検証エラー時のメッセージ

Arduino IDEの右下にあるアイコン 一度クリックすると表示されます。画面を大きくしてスケッチ全体を少しでも多く見たい時や、この メッセージ画面が不要なとき等に活用してください。

Ubuntuの場合、ModuleNotFoundError: No module named 'serial'のメッセージが出力された ときはビルドができていません。下記のコマンドをターミナルで実行し、ビルドに必要なアプリをイ ンストールしてください。

\$ sudo apt-get install python-pip \$ pip install pyserial \$ pip install esptool



#### 書き込み

スケッチにエラーが無いことを確認したら、次の手順で書き込み前の準備をします。

- 1. 本体にバッテリが繋がっていることを確認します。
- 2. PCと本製品をUSBケーブルで接続します。
- 3. 本製品のモード切替スイッチをFWと書かれている方向にスライドします。
- 4. Pi:Co V2の電源スイッチをON側にスライドします。
- 5. 本製品のリセットボタンを押します。



モード切替スイッチを操作し、USBケーブルを接続した状態



本製品をPCと接続した後、以下の手順でボードとポートを設定します。本製品の電源をOFFし た後や、リセットボタンを押した後にも再度設定してください。

メニューバーの下に表示されているボード名(ESP32S3 Dev Module)をクリックし、一番下の" 他のボードとポートを選択…"を開きます。



ボード名->他のボードとポートを選択...を開く

ボードの検索欄に"dev"を入力し、"ESP32S3 Dev Module"にチェックが入っていることを確認し ます。チェックが入っていない場合は、ESP32S3 Dev Moduleをクリックします。ポート選択欄で は本製品を接続したUSBポートを選択します。Windowsの場合はCOMX、Linux(Ubuntu)の場 合は/dev/ttyUSBX、macOSの場合はusbmodemXXXXX Serial Port(USB)という名称です。選 択したら"OK"をクリックします。

他のボードとポートを選択										
スケッチを書き込みたい場合には、ボードとポートの両方を選択してください。 ボードのみを選択した場合、コンパイルはできますが、スケッチの書き込みはできません。										
ボード		ポート								
dev	۹									
ESP32H2 Dev Module		COM4 Serial Port (USB)	~							
ESP32S2 Dev Module	- 1	COM1 Serial Port								
ESP32S3 Dev Module	~									
ESP32S3 Dev Module Octal (WROOM2)										
Hornbill ESP32 Dev										
Lion:Bit Dev Board										
		キャンセル	ОК(О)							

ボードとポートの設定



ポート設定後、 をクリックしてスケッチを書き込みます。 下記のように表示されたら書き込み完了です。



書き込み完了時の画面表示

Linux(Ubuntu)PCで書き込みに失敗する時は、書き込み前にターミナルで次のコマンドを実行し、USBポートの使用権限を変更します。

```
$ sudo chmod 777 /dev/ttyACM0
```

永続的にUSBポートの使用権限を変更する場合は、次のコマンドを実行し、PCを再起動します。

```
# 下記を実行した後にPCを再起動すること
$ sudo usermod -aG dialout $USER
```

スケッチの書き込み完了後、本製品のモード切替スイッチをRUN側にスライドし、リセットボタンを押します。

Pi:Co V2前方のLEDが0.5秒点滅、0.5秒消灯を繰り返します。





#### STM32ボード情報の追加

スケッチを本製品に書き込めるように、STM32のボード情報をArduino IDEに追加します。メ ニューバーのファイル->基本設定...を選択します。画面下部の"追加のボードマネージャのURL" に

https://github.com/stm32duino/BoardManagerFiles/raw/main/package\_stmicroelectronics\_index.json

を追加し"OK"をクリックします。



		設定	ネットワーク			
スケッチブックの場所:						
c:\Users\maaok\OneDrive\Docu	ments\Arduin	)			(	参照
<ul> <li>スケッチ内のファイルを表示</li> <li>エディターのフォントサイズ:</li> </ul>	14					
インターフェイスのスケール:	☑ 自動 10	%				
配色テーマ:	Light	~				
エディター言語:	日本語	✓ (Relo	ad required)			
より詳細な情報を表示する	マコンパイ	レ□書き込み				
コンパイラの警告	なし	×				
<ul> <li>✓ 書き込み後にコードを検証す</li> <li>✓ 自動保存(U)</li> <li>□ エディターのクイックサジェ</li> </ul>	る					
 追加のボードマネージャのURL:	https://githu	.com/stm32duir	no/BoardManag	erFiles/raw/main/pa	kage_stmicroel	Ð

追加ボードマネージャーのURLに追加した状態

URLを追加したら、STM32ライブラリやツールをダウンロードしてインストールします。メニューバーのツール->ボード->ボードマネージャ...を選択します。

🔤 ske	etch_ma	r5b   A	Ardui	ino IDE 2.3.4			_		×	
ファイル	編集	スケッチ	F .	ソール ヘルプ(H)						1
	Ð		Γ	自動整形	Ctrl+T				$\checkmark$	۰ <b>©</b> ۰۰
	akata	b mo	-6	スケッチをアーカイブする						
	skelt	.n_ma	cıı	ライブラリを管理	Ctrl+Shift+I					
		1	va	シリアルモニタ	Ctrl+Shift+M		n once:			
딉		3		シリアルプロッタ			in once.			
		4	}	Firmware Updater						
		6	va	SSLルート証明書を書き込み						
		7		ボード		•	ボードマネージャ	Ctrl	+Shift+l	3
0		9	}	ポート		×	esp32			•
	1	9		Reload Board Data			STM32 MCU based boards	;		•
Q				ボード情報を取得						
				ブートローダを書き込む						

ボードマネージャ...を選択した状態

左側にあるボードマネージャの検索欄に"stm32"と入力すると"STM32 MCU Based boards by STMicroelectronics"が表示されます。2025年4月時点では2.10.1が最新です。STM32 MCU Based boards by STMicroelectronicsのインストールをクリックします。





ボードマネージャ画面の検索欄に"stm32"を入力した状態

Arduino IDEは STM32以外のマイコンにも対応しているため、スケッチを書き込む際はSTM32 のボードを選択します。メニューバーのツール->ボード->STM32 MCU based boards ->Nucleo-32を選択します。"ボード"と"ポート"が同じような文字で上下にあるため間違えないよう に選択してください。

🔤 sket	ch_mar10a   Ard	luino IDE 2.3.4						
ファイル	編集 スケッチ	ツール ヘルプ(H)						
	<b>→</b> 🔊 [	自動整形	Ctrl+T					
	+ 1-7+	スケッチをアーカイブする						
	M- PXA-:	ライブラリを管理	Ctrl+Shift+I					
	stm32	シリアルモニタ	Ctrl+Shift+M		to run once:			
	タイプ: 全	シリアルプロッタ						
-		Firmware Updater						
	STM32 MC STMicroele	SSLルート証明書を書き込み						
	2.9.0 install	ボード: "ESP32S3 Dev Module"		Þ	ボードマネージャ Ctrl+	Shift+B		
÷>	Boards inclu boards. Mid	ポート	1	Þ	esp32		Þ	
$\sim$	STM32L5 se	Reload Board Data			STM32 MCU based boards		Þ	Nucleo-144
X	a开和11月辛收	ボード情報を取得						Nucleo-64
	2.9.0 🗸	USB CDC On Boot: "Disabled"	1	۲				Nucleo-32
		CPU Frequency: "240MHz (WiFi)"		۲				Discovery
		Core Debug Level: "None"		۲				Eval
	STM8 MCI STMicroele	USB DFU On Boot: "Disabled"		۲				STM32MP1 series coprocessor
	Boards inclu	Erase All Flash Before Sketch Upload: "Disabled"	1	۲				Generic STM32C0 series
	Nucleo_8S2( 詳細情報	Events Run On: "Core 1"	1	۲				Generic STM32F0 series

ツール->ボード->STM32 MCU based boards->Nucleo-32を選択

次にNuleo-32から使用するボードを選択します。ツール -> Board part numberでNucleo F303K8を選択します。



🔤 ske	etch_mar10a   Ard	uino IDE 2.3.4		
ファイル	編集 スケッチ	ツール ヘルプ(H)		
Ø	€ 🖗	自動整形	Ctrl+T	
	ボードマネー: stm32	スケッナをアーカイフする ライブラリを管理 シリアルモニタ	Ctrl+Shift+I Ctrl+Shift+M	{
包	タイプ: 全	シリアルプロッタ		secup code nere, i
	STM32 MC STMicroele 2.9.0 install Boards inclu boards, Mid STM32L5 se 詳細情報 2.9.0 ◆	Firmware Updater SSLルート証明書を書き込み ボード: "Nucleo-32" ポート Reload Board Data ボード情報を取得 Debug symbols and core logs: "None	•	main code here, to
	STM8 MCI STMicroele Boards inclu Nucleo_852( 詳細情報 1.0.0 ✔	Optimize: "Smallest (-Os default)" Board part number: "Nucleo F303K8" C Runtime Library: "Newlib Nano (def Upload method: "Mass Storage" USB support (if available): "None" U(S)ART support: "Enabled (generic 'S USB speed (if available): "Low/Full Spe ブートローダを書き込む	ault)"	Nucleo F031K6 Nucleo F042K6 Vucleo F303K8 Nucleo G031K8 Nucleo G431KB Nucleo L031K6 Nucleo L412KB Nucleo L412KB

ツール->Board part Number->Nucleo F303K8を選択

以上で、STM32 Nucle F303K8のビルド環境設定完了です。

#### 書き込みツールSTM32CubeProgrammerのインストール

Arduino IDEからNuleoボードにプログラムを書き込むにはSTMicroelectronics社の STM32CubeProgrammerに含まれているSTM32\_Programmer\_CLIが必要になります。 STM32CubeProgrammerは<u>https://www.st.com/ja/development-tools/stm32cubeprog.html</u> からダウンロードします。

使用するOSに合うソフトウェアを"最新バージョンの取得"をクリックしてダウンロードします。

#### ソフトウェア入手

	製品型番	概要	÷	ダウンロード	÷	すべてのバージョン
+	STM32CubePrg-Lin	STM32CubeProgrammer software for Linux		最新バージョンを取得		バージョンを選択
+	STM32CubePrg-Mac	STM32CubeProgrammer software for Mac		最新バージョンを取得		バージョンを選択
+	STM32CubePrg-W32	STM32CubeProgrammer software for Win32		最新バージョンを取得		バージョンを選択
+	STM32CubePrg-W64	STM32CubeProgrammer software for Win64		最新バージョンを取得		バージョンを選択

ライセンス契約ついて確認し"承諾します"をクリックします。

Please indicate your acceptance or NON-acceptance ACCEPT <sup>®</sup> as indicated below in the media.	by selecting "I ACCEPT" or "I DO NOT
BY INSTALLING COPYING, DOWNLOADING, ACCE SOFTWARE PACKAGE OR ANY PART THEREOF ( DOCUMENTATION) FROM STMICROELECTRONIC BRANCH AND/OR ITS AFFILIATED COMPANIES (S RECIPIENT, ON BEHALF OF HIMSELF OR HERSEI BY WHICH SUCH RECIPIENT IS EMPLOYED AND/ BOUND BY THIS SOFTWARE PACKAGE LICENSE	ESSING OR OTHERWISE USING THIS AND THE RELATED S INTERNATIONAL N.V, SWISS TMICROELECTRONICS), THE LF, OR ON BEHALF OF ANY ENTITY OR ENGAGED AGREES TO BE AGREEMENT.
Under STMicroelectronics' intellectual property rights terms for any third-party software incorporated in this Source Terms (as defined here below), the redistribut binary forms of the software package or any part ther permitted provided that the following conditions are m	and subject to applicable licensing software package and applicable Open ion, reproduction and use in source and eof, with or without modification, are net:
<ol> <li>Redistribution of source code (modified or not) list of conditions and the following disclaimer.</li> </ol>	must retain any copyright notice, this
<ol> <li>Redistributions in binary form, except as embe microprocessor device manufactured by or for update for such device, must reproduce the ab</li> </ol>	dded into microcontroller or STMicroelectronics or a software ove copyright potice, this list of
Additional License Terms for STM32CubeProgramm	ner 2 16 0

MySTアカウントでログインします。アカウントを所持していない場合は新規作成してください。





登録してあるメールアドレスとパスワードを入力しログインをクリックします。

mySTログイン	新規に登録される方
メールアドレスとパスワードを入力し、ログインしてください	mySTのパーソナライズ機能
メールアドレス	◦ イベントへの参加 ◦ ニュースレター配信 ◦ オンライン・サポート利用
パスワード	<ul> <li>STコミュニティへの参加</li> <li>オンライン設計ツール利用</li> <li>ソフトウェア・ダウンロート</li> <li>毎億世ンプル送立</li> </ul>
□ メールアドレスを記憶 🕕	<ul> <li>         ・</li></ul>
<b>ログイン</b> パスワードを忘れた場合	新規登録

ログインが成功するとダウンロードが開始されます。

STM32CubeProgrammerインストーラの起動方法はOS毎に異なりますが、インストール手順はほとんど同じです。ここではインストーラーの起動後はWindowsの画面で説明します。

Windowsでのインストーラ起動方法

ダウンロードしたen.stm32cubeprog-win64-v2-19-0.zipファイルを右クリックし"すべてを展開"を 選択し展開します。展開されたフォルダにあるSetupSTM32CubeProgrammer\_win64.exeをダブ ルクリックで実行します。



"はい"をクリックします。



#### Linux(Ubuntu)でのインストーラ起動方法

ダウンロードしたen.stm32cubeprg-lin-v2-19-0.zipファイルを右クリックで展開します。展開されたフォルダにあるSetupSTM32CubeProgrammer-2.19.0.linuxをダブルクリックで実行します。



macOSでのインストーラ起動方法

ダウンロードしたen.stm32cubeprg-mac-v2-19-0.zipファイルをダブルクリックで展開し、 SetupSTM32CubeProgrammer-2.19.0をダブルクリックします。



"開く"をクリックします。

#### インストーラの操作手順(OS共通)



Nextをクリックします。

Nextをクリックします。







Yes No



#### YESをクリックします。 I haveにチェックを入れてNextをクリックします。

STH03CubeProgram	nner installation Wicard	= 0 X	Fig STM32CubeProgrammer Installation Wizard	- 0	×
STN020.deProgramme Disp 6 of 9	Components selection	STM 32 Gabe Programmer	STM32CubeProgrammer Package installation Step 7 of 9	STM32 CubeProgr	l grammer
	Todar Free meganicus en esta tentali Nece Tanad and and en emploided B STMECOuchongeneme STMECF satudfuntageCouche STMECF satudfuntageCouche Description Genergton Control from	1980 17 MMC 8173 PMC 81737 MG	Pack Installation progress: ChProgram Files/STMccodestorrics/STMcCodes/STMCCodes	iniçx64.dll	
Phinesectories -	Nextをクリックします。	titet te ⊕™ener (€ter) €ur 112	℠Merceledaries	Next @	Quit

(windowsのみ)

Warning

(windowsのみ)



en 🛊 Nert O Cuit



次へをクリックします。

### 完了をクリックします。

57MS2CubeProgrammer Installation Wicard	- 0 ×	TMUCubeProgram	nner Indalletion Villaard	- 0 X
STM32Galafhogrammer Pauloge Inclution Ship 7 at 1		STREECuturProgrammer Disp 5 of 3	Bartosh integ	STMC: 0 CubeProgrammer
Fact including symptoms Factors Factor			Create startistis in the Dat-Menu     Create stafficient distributions for distribution     Create all Property distributions     Created     Cre	orade shortsof for Createst are Ordinaers
11Mandrawa	e en en	\$756-poekentorios	(17%) value har existence of 17%) (12%)	fordat 💓 1944

Nextをクリックします。





Doneをクリックします。

デバイスドライバのインストール(WIndowsのみ)

ここでは、プログラムの書き込みに使用するデバイスドライバがPCにインストールされているか を確認します。USBケーブルでNucleoボードとPCを接続します。 Nucleoボードはトレーニングトレーサーに載せたままケーブルを繋いでください。







スタートボタンの。 アイコンをクリックして、「デバイスマネージャー」と入力し、検索結果にある デバイスマネージャーをクリックします。



ポート(COMとLPT)の項目にSTMicroelectronics STLink Virtural COM Port(COMX)があり、 ユニバーサルシリアルバスデバイスの項目にST-Link Debugがあればデバイスドライバをインス トールする必要はありません。



ほかのデバイスまたは不明なデバイスの中に下図のような黄色の三角形で警告が出ている場合があります。この場合はデバイスドライバのインストール作業が必要です。





デバイスドライバはSTMicroelectronics社の

<u>https://www.st.com/ja/development-tools/stsw-link009.html</u>から最新バージョンをクリックしてダウンロードします。

```
ソフトウェア入手
```

	製品型番	•	最新バージョン	$\frac{\mathbb{A}}{\mathbb{V}}$	ダウンロード	÷
+	STSW-LINK009		2.0.2		最新バージョンをダウンロード	

"ソフトウェア入手"をクリックします。ライセンス契約の確認画面が表示されるので"同意"をクリックします。

ダウンロードしたen.stsw-link009.zipを展開します。 先程開いたデバイスマネージャーで警告が出ている箇所をダブルクリックして開きます。

ST-Link D	)ebugのプロ	パティ		$\times$		
全般	ドライバー	詳細	イベント			
1	ST-Lin	Debug				
	デバイス	の種類:	ほかのデバイス			
	製造元		不明			
	場所:		0004.0000.0003.003.000.000.000.000.000			
デバー	「スの状態ー					
1 to	のデバイスのト	・ライバーが	インストールされていません。 (コード 28)	^		
20	Dデバイスと <u>E</u>	互換性のあ	るドライバーがありません。			
20 10,	このデバイス用のドライバーを検索するには、[ドライバーの更新]をクリックしてください。					
			ドライバーの更新(U)			
			ОК <b>‡</b> †	ンセル		

プロパティが開いたら"ドライバーの更新"をクリックします。





ドライバーの検索方法という画面が開いたら、"コンピューターを参照してドライバーソフトウェアを検索"をクリックします。



"コンピューター上のドライバーを参照します。"という画面で参照をクリックし、先程ダウンロートして展開したen\_stsw-link009があるフォルダを選択します。





"Windowsセキュリティ"という画面で"インストール"をクリックします。

		$\times$
¢	■ ドライバーの更新 - ST-Link Debug	
	ドライバーが正常に更新されました	
	このデバイス用のドライバーがインストールされました	
	STMicroelectronics STLink dongle	

"正常に更新されました"と表示されたら"閉じる"をクリックします。

TRT CORPORATION

#### サンプルスケッチの準備

webブラウザで<u>https://github.com/rt-net/TrainingTracer\_V2\_Samples</u> にアクセスし、トレーニングトレーサーのサンプルスケッチファイルをダウンロードします。

ページ内の"Code"から"Download ZIP"をクリックすることでダウンロードできます。ダウンロード したzipファイルは展開してください。

TrainingTracer_V2_Samples	Private		© Undeh (
Ƴ main → ၾ 1 Branch ⓒ 0 Tags	Q Go to fil	e t	Add file - <> Code -
YusukeKato サンブルブログラムをチェッ?	クするGitHub Workflowを追	Local	Codespaces
github/workflows	サンプルプログラム	E Clone	0
Arduino_Exercises	サンプルプログラム	HTTPS SSH GitHu	b CLI
Arduino_Sample	サンプルプログラム	git@github.com:rt-net/	/TrainingTracer_V2_San
STM32CubeIDE_Sample	サンプルプログラム	Use a password-protected	d SSH key.
STM32CubeIDE_file	サンプル	Download ZIP	
C .clang-format	サンプルプログラム	をチェックするGitHub Work	flowを追 20 hours ago
	Initial commit		vactarylau

Code->Download ZIPを選択してサンプルスケッチをダウンロード

ここではNucleoのボード上にあるLEDを点灯させるサンプルスケッチ(STEP1)を例にとってスケッチのビルドと書き込み方法を説明します。

本製品にはUSB経由でスケッチを書き込みます。



ダウンロードしたサンプルスケッチ内のArduino\_Sample/STEP1/STEP1.inoファイルをダブル クリックで開きます。または、Arduino IDEを起動し、メニューバーのファイル->開く...をクリックして STEP1.inoを開きます。

🥯 sketch_mar13a   Ar	duino IDE 2.3.4	
ファイル 編集 スケッチ	ツール ヘルプ(H)	
新規スケッチ	Ctrl+N	-
新規クラウドスケッチ	Alt+Ctrl+N	
開く	Ctrl+O	
最近使った項目を開く	•	etup code here.
スケッチブック	•	,
スケッチ例	•	
閉じる	Ctrl+W	
Save	Ctrl+S	ain code here,
を前たけけて保方 ファイ	┌┿┧ュႽトi╫ュႽ ゙ル->開くを選択	する

スケッチファイルを開く時に以下のメッセージが出た場合は、スケッチファイル名とフォルダ名が 一致していない、またはフォルダが無いため新しくフォルダを作って良いか、ということを確認して います。Arduino IDEの仕様上、メインとなる"ファイル名"と"フォルダ名"が同じでないときにこの メッセージが表示されます。ファイル名やフォルダ名を変更する場合はそれぞれ同じ名前にしてく ださい。

移動		$\times$
1	ファイル"STEP1.ino"は、"STEP1"という名前のスケッチフォルダの中にあることが必要です。 このフォルダを作成し、ファイルを移動させ、継続しますか?	
	→ キャンセル	
	$\rightarrow OK(O)$	

ファイル名とフォルダ名が一致しない場合の確認画面



サンプルスケッチを開くと以下のような画面が表示されます。

```
🔤 STEP1 | Arduino IDE 2.3.4
ファイル 編集 スケッチ ツール ヘルプ(H)
                ボードを選択
      \rightarrow
          -
      STEP1.ino
         1
              /*
          2
               *
                    Arduino Sample/STEP1/STEP1.ino
               *
          3
         4
               *
                   Copyright(C) 2024 RT Corporation <support@rt-net.jp>
         5
               *
                     All rights reserved.
               *
          6
         7
               *
                   License: Apache License, Version 2.0
               *
                     https://www.apache.org/licenses/LICENSE-2.0
         8
 0
               *
         9
              */
        10
        11
        12
              #define LED_PIN D13
        13
        14
             void setup()
        15
              {
        16
                // put your setup code here, to run once:
                pinMode(LED_PIN, OUTPUT);
        17
        18
                pinMode(D12, OUTPUT);
        19
               digitalWrite(D12, LOW);
        20
              }
        21
             void loop()
        22
        23
             {
                // put your main code here, to run repeatedly:
        24
                digitalWrite(LED_PIN, HIGH); //LEDを点灯
        25
        26
                delay(1000);
                                              //1秒待つ
               digitalWrite(LED_PIN, LOW); //LED消灯
        27
               delay(1000);
                                              //1秒待つ
        28
         29
              }
```

STEP1.inoを開いた状態



ボードが"Nucle-32"になっていないときは、メニューバーのツール->ボード->STM32 MCU based boards ->Nucleo-32を選択してください。また Board part numberがNucleo F303K8に なっていることも確認してください。Board part numberはツール-> Board part numberで選択で きます。

k	編集 スケ	7F	ツール ヘルプ(H)			_	
	€ 6		自動整形	Ctrl+T			
-	al at at		スケッチをアーカイブする				
	sketch_r	hayı	ライブラリを管理	Ctrl+Shift+I			
	2	ve	シリアルモニタ	Ctrl+Shift+M			
	3		シリアルプロッタ				
	4	)	Firmware Updater				
D	6	ve	SSLルート証明書を書き込み				
	7		ポード: "Nucleo-32"		,		
>	9	3	ポート		٠		
	10	1	ポード情報を取得				
ζ			Debug symbols and core logs: "None"				
			Optimize: "Smallest (-Os default)"		,		
			Board part number: "Nucleo F031K6"			~	Nucleo F031K6
			C Runtime Library: "Newlib Nano (default)"				Nucleo F042K6
			Upload method: "Mass Storage"				Nucleo F303K8
			USB support (if available): "None"				Nucleo G031K8
			U(S)ART support: "Enabled (generic 'Serial')"		×		Nucleo G431KB
			USB speed (if available): "Low/Full Speed"		×		Nucleo L031K6
			ブートローダを書き込む				Nucleo L412KB
							Nucleo L432KC

ツール-> Board part number->Nucleo-F303K8を選択



ビルド

左上にあるアイコンで"検証"を実行しスケッチをビルドします。スケッチの記述に問題がなければ以下のようなメッセージが表示されます。

出力
最大65536バイトのフラッシュメモリのうち、スケッチが12484バイト(19%)を使っています。 最大12288バイトのRAMのうち、グローバル変数が1244バイト(10%)を使っていて、ローカル変数で11044バイト使うことができます。
検証に問題が無いときのメッセージ
スケッチの記述にエラーがあった場合は、赤文字でエラー箇所が表示されます。
шл ☴ 읍
<pre>"C:\Users\\maaok\\AppData\Local\\Arduino15\\packages\\STMicroelectronics\\tools\\xpack-arm-none-eabi-gcc\\12.2.1-1.2. "C:\Users\\maaok\\AppData\Local\\Arduino15\\packages\\STMicroelectronics\\tools\\xpack-arm-none-eabi-gcc\\12.2.1-1.2. C:\Users\\maaok\Downloads\TrainingTracer_Samples-master\TrainingTracer_Samples-master\Arduino_Sample\STEP1\STEP1.ino: I C:\Users\maaok\Downloads\TrainingTracer_Samples-master\TrainingTracer_Samples-master\Arduino_Sample\STEP1\STEP1.ino: I C:\Users\maaok\Downloads\TrainingTracer_Samples-master\TrainingTracer_Samples-master\Arduino_Sample\STEP1\STEP1.ino: I C:\Users\maaok\Downloads\TrainingTracer_Samples-master\TrainingTracer_Samples-master\Arduino_Sample\STEP1\STEP1.ino: 16     [16] pinMode(LED1_Pin,OUTPUT);     [1</pre>
Compilation error: 'LED1_Pin' was not declared in this scope; did you mean 'LED_Pin'?

検証エラー時のメッセージ

Arduino IDEの右下にあるアイコン 一度クリックすると表示されます。画面を大きくしてスケッチ全体を少しでも多く見たい時や、この メッセージ画面が不要なとき等に活用してください。



書き込み

書き込みにはCOMポートとSWDを使用します。

COMポートはツール->ポートで設定します。COM番号はデバイスマネージャーで確認できます。

Windowsの場合はCOMX、Linux(Ubuntu)の場合は/dev/ttyUSBX、macOSの場合は usbmodemXXXXX Serial Port(USB)という名称です。

	ソール ヘルプ(H)			
Γ	自動整形	Ctrl+T		
١.	スケッチをアーカイブする			
Į.	ライブラリを管理	Ctrl+Shift+I		
	シリアルモニタ	Ctrl+Shift+M		
*	シリアルプロッタ			
8	Firmware Updater			t-net.jp>
*	SSLルート証明書を書き込み			
*	ボード: "Nucleo-32"		۲	~
*	ポート		۲	シリアルポート
*	ボード情報を取得			COM1
n	Debug symbols and core logs: "None"		۲	COM5
	A			

SWDはツール->Upload method->STM32CubeProgrammer(SWD)を選択します。

スケッ	Ŧ	ツール ヘルプ(H)					
	Г	自動整形	Ctrl+T				
		スケッチをアーカイブする					
'1.in	0	ライブラリを管理	Ctrl+Shift+I		P		
		シリアルモニタ	Ctrl+Shift+M				
i.	*	シリアルブロッタ					
ł	1	Firmware Updater			t-	net	t.jp>
5		SSLルート証明書を書き込み					
!	1	ポード: "Nucleo-32"		٠			
1	1	ボート		٠	9		
•		ポード情報を取得					
ł.	in	Debug symbols and core logs: "None"		٠			
1		Optimize: "Smallest (-Os default)"		٠			
1	VC	Board part number: "Nucleo F303K8"		٠			
5		C Runtime Library: "Newlib Nano (default)"		٠	١.		
!		Upload method: "Mass Storage"		٠	•	~	Mass Storage
1	}	USB support (if available): "None"		٠			STM32CubeProgrammer (SWD)
)	va	U(S)ART support: "Enabled (generic 'Serial')*		٠			STM32CubeProgrammer (Serial)
1		USB speed (if available): "Low/Full Speed"		٠			STM32CubeProgrammer (DFU)



スケッチにエラーが無いことを確認したら、次の手順で書き込み前の準備をします。

- 1. ポート番号とUpload methodが正しいか確認します。
- 2. 本体に乾電池をセットしていることを確認します。
- 3. PCと本製品をUSBケーブルで接続します。
- 4. タイヤを浮かせます。(書き込み実行時に左のモータが回転するため)
- 5. 電源スイッチをON側にスライドします。



USBケーブルを接続した状態

正しく設定されているとボート名の左にUSBのマーク · Nucleo-32 · が 追加されます。スケッチの書き込み以外に、PCとハードウェアシリアル通信をするときも、ポート 設定が必要です。

ポート設定後、 をクリックしてスケッチを書き込みます。 下記のように表示されたら書き込み完了です。

#### 出力



書き込み完了時の画面表示



Nucleoボード上にあるLEDが0.5秒点滅、0.5秒消灯を繰り返します。



#### 書き込み時のエラー対処方法

ST-Linkのファームウェアが古いと以下のメッセージが出ます。このメッセージが表示された時は、STlinkのファームウェアのアップデートを行ってください。

	STM32CubeProgrammer v2.3.0	
Error: Error: Error:	Old ST-LINK firmware version. Upgrade ST-LINK firmware Old ST-LINK firmware version. Upgrade ST-LINK firmware Old ST-LINK firmware!Please upgrade it.	

Error: No debug probe detectedの表示が出た時は、PCとNucleoボードがUSBケーブルで正常に接続されていません。USBケーブルを接続し直し、再度書き込みをしてください。



Error: No STM32 target foundの表示が出た時は、本製品の電源が入っていません。電源をON側にスライドしたあと再度書き込みをしてください。

```
ST-LINK SN : 0667FF303032424257015845
ST-LINK FW : V2J44M29
Board : NUCLEO-F303K8
Voltage : 3.27V
Error: No STM32 target found! If your product embeds Debug Authe
Failed uploading: uploading error: exit status 1
```



#### ST-Linkファームウェアのアップデート

以下のページからST-Linkのファームウェアをダウンロードします。 https://www.st.com/ja/development-tools/stsw-link007.html

#### Windowsでのファームウェアアップデート方法

ダウンロードしたファイルを展開し、stsw-linl007/windows フォルダにあるST-LinkUpgrade.exeをダブルクリックで起動します。

📣 ST-LinkUpgrade	2024/04/03 10:27	アプリケーション	1,253 KB
STLinkUSBDriver.dll	2024/04/03 10:27	アプリケーション拡張	115 KB

📣 ST-Link Upgrade	×
Device Connect Current Firmware: Version: Type:	STM32
Upgrade to Firmware:	Yes >>>>

NucleoボードとPCをUSBで接続し、電源スイッチをスライドした後、"Device Connect"をクリックします。



VersionやTypeが表示されたら"Yes>>>>"をクリックします。





Upgrade is successfulと表示されたら、この画面を閉じて書き込みを再開します。



Linux(Ubuntu)でのファームウェアアップデート方法

ダウンロードしたenstsw-link007-v3-16-7.zipをダブルクリックで展開します。ターミナルを起動し、展開したswst-link007/AllPlatformsに移動します。

java -jar	STLinkUpgrade.jar			
	-			
		STLinkUpgrade 3.14.5	(_) (I) (X)	
		▼ Refresh device list	<u></u>	
	Open in update mode			
	ST-Link ID:		Releasing your creativity	
	Current Firmware:			
	Type:			
	Version:			
	Update to Firmware:			
	Upgrade			

NucleoボードとPCをUSBで接続し、電源スイッチをスライドした後、"Refresh Device list"をクリックします。

	STLinkUpgrade 3.14.5	- •	×
ST-LINK/V2-1	Refresh device list	1	2
Open in update mode	STM2	-	
ST-Link ID: 0667FF303032424257015845	Releasing your creativity	Y S	
Current Firmware:			
Type:			
Version: Unknown			
Update to Firmware: Unknown			
Upgrade			

"Open in update mode"をクリックします。



STLinkUpgrade 3.14.5		×
ST-LINK/V2-1 Refresh device list	And a	4
Open in update mode	-	9
ST-Link ID: 0667FF303032424257015845		
Current Firmware:		
Type: STM32 Debug+Mass storage+VCP		
Change Type		
Version: V2J45M30		
Update to Firmware: V2J44M29 STM32 Debug+Mass storage+VCP		
Upgrade		

VersionやTypeが表示されたら"Upgrade"をクリックします。

	STLinkUpgrade 3.14.5	- O X
ST-LINK/V2-1	▼ Refresh device list	
Open in update mode		S S S S S S S S S S S S S S S S S S S
ST-Link ID: 0667FF3030324242570158	345	Releasing your creativity
Current Firmware:		
Type: STM32 Debug+Mass storage+	VCP	
Version: V2J44M29		
Update to Firmware: V2J44M29 STM	132 Debug+Mass storage+VCP	
Upgrade		
Upgrade successful.		

"Upgrade is successful"と表示されたら、この画面を閉じて、書き込みを再開します。



macOSでのファームウェアアップデート方法

ダウンロードしたenstsw-link007-v3-16-7.zipをダブルクリックで展開します。ターミナルを起動し、展開したswst-link007/AllPlatformsに移動します。

java -jar STLinkUpgrade.jar

このメッセージが表示された場合は、キャンセルを押します。



システムのプライバシーとセキュリティの設定項目で"このまま許可"をクリックします。





プライバシーとセキュリティ
プライバシーとセキュリティがシステム設定を 変更しようとしています。
許可するにはパスワードを入力してください。
<b>許可するにはパスワードを入力してください。</b> ユーザ名
許可するにはパスワードを入力してください。 ユーザ名 パスワード

PCのユーザ名とパスワードを入力して設定を変更します。再度、ターミナルに戻り、javaのコマンドを実行します。

? "libSTLinkUSBDriver.dylib"	
の開発元を検証できません。 開いてもよろしいですか?	
このアプリケーションを開くことによって、 システムのセキュリティが無効になり、コン ピュータと個人情報がマルウェアにさらされ る場合があります。その結果、マルウェアに よって、Mac やプライパシーに損害を受ける 可能性があります。	
このファイルは "Safari" により今日の16:25 にダウンロードされました。	
開く	
ゴミ箱に入れる	
キャンセル	

先程と似たようなメッセージが表示されますが、今度は"開く"をクリックします。





再びキャンセルし、システムのプライバシーとセキュリティの設定項目で"このまま許可"をクリックします。再度、ターミナルに戻り、javaのコマンドを実行します。

•••	STLinkUpgrade 3.15.6
ST-LINK/V2-1	Refresh device list
Open in update mode	
ST-Link ID: 0667FF303032424257015845	Releasing your creativity
Current Firmware:	
Type:	
Version: Unknown	
Update to Firmware: Unknown	
Upgrade	

NucleoボードとPCをUSBで接続し、電源スイッチをスライドした後、"Open in update mode"を クリックします。"Open In update mode"をクリックできない時は、"Refresh device list"をクリック するとクリックできるようになります。



#### Arduino開発環境構築マニュアル

•••	STLinkUpgrade 3.15.6
ST-LINK/V2-1	Refresh device list
Open in update mode	
ST-Link ID: 0667FF303032424257015845	Releasing your creativity
Current Firmware:	
Type: STM32 Debug+Mass storage+VCP	
Change Type	
Version: V2J44M29	
Update to Firmware: V2J45M30 STM32 Debug+	Mass storage+VCP
Upgrade	

VersionやTypeが表示されたら"Upgrade"をクリックします。

• • •	STLinkUpgrade 3.15.6
ST-LINK/V2-1	Refresh device list
Open in update mode	
ST-Link ID: 0667FF303032424257015845	Releasing your creativity
Current Firmware:	
Type: STM32 Debug+Mass storage+VCP	
Version: V2J45M30	
Update to Firmware: V2J45M30 STM32 Debug+	Mass storage+VCP
Upgrade	
Upgrade successful.	

"Upgrade is successful"と表示されたら、この画面を閉じて、書き込みを再開します。



## 使用しているツール、OSSのバージョン

本書で使用するツールやOSSのバージョンは以下のとおりです。

ツールやOSS	バージョン	URL
Arduino IDE	2.3.6	https://www.arduino.cc/en/software
Arduino core for the ESP32	3.2.0	https://github.com/espressif/arduino-esp 32/releases
Arduino core support for STM32 based boards	2.10.1	https://github.com/stm32duino/Arduino_ Core_STM32/releases
STM32CubeProgrammer	2.19.0	https://www.st.com/ja/development-tools/ stm32cubeprog.html
STLink boards firmware upgrade	3.16.7	https://www.st.com/ja/development-tools/ stsw-link007.html
STLink USB driver (windows only)	2.0.2	https://www.st.com/ja/development-tools/ stsw-link009.html

#### 本書で使用するツールやOSSのバージョン情報



# 改訂履歴

発行日 (YY/MM/DD)	版数	改訂内容
24/06/12	1.0	新規作成
24/09/20	1.1	Training Tracer Ver.2のGitHubリポジトリのURLに更新
25/06/01	1.2	対象PiCoClassic3からPi:CoClassic4に変更 OSSのバージョンを更新

## **Copyright**・知的財産権について

弊社は、本製品および本製品に関連して弊社が制作したソースファイル、ディレクトリ、実行可 能ファイル、データ、開発ツールおよびその他の資料(以下「弊社資料」といいます)にかかる特 許権、実用新案権、意匠権、著作権、ノウハウ、その他の技術および知的財産に関する一切の 権利を留保するものとします。本書は、弊社の商標、商号、役務商標、商品名、ロゴの使用を許 諾するものではありません。ただし、本製品および弊社資料の説明または記述に合理的に必要 な範囲において当該商標等を使用する場合は、この限りでないものとします。なお、本製品およ び弊社資料に付された商品識別番号、商標、登録商標、コピーライト、その他の注意事項は除去 しないようお願いします。

All the company and product names in this document are trademarks or registered trademarks of their respective companies.

All the documents, photos, and illustrations are copyrighted and protected by the copyright law of Japan and overseas. All the contents in this document are not allowed to be uploaded to any public or local area networks such as the Internet without permission from RT Corporation.

