RAK3172 Module Low Level Development Reference

Overview

The RAK3172 module uses RF System-on-Chip IC from STMicroelectronics, specifically the STM32WL series, which can be used on different LPWAN IoT devices.

STM32WL microcontrollers feature a sub-GHz radio based on Semtech SX126x to meet the requirements of a wide range of Low-Power Wide Area Network (LPWAN) wireless applications in industrial and consumer Internetof-Things (IoT). The specific STM32WL microcontroller used in RAK3172 is the STM32WLE5CCU6.

While RAK3172 has a built-in default FW with a set of AT commands that can be interfaced to an external host like other microcontrollers, it can also be used by developing custom firmware directly on its chip using the STM32WL SDK from STMicroelectronics. Doing this approach will reduce the overall cost of the device because there will be no need for an external microcontroller but with the extra software development effort.

This guide will illustrate how to generate custom firmware for the STM32WLE5CCU6, which is inside the RAK3172 module. It supports two STM32WL SDK versions - v1.0.0 and v1.2.0.

- STM32CubeIDE guide with STM32WL SDK v1.0.0
- STM32CubeIDE guide with STM32WL SDK v1.2.0

Guide on Using STM32WL SDK Using STM32CubeIDE

Installation of STM32Cube IDE

1. Download the STM32Cube IDE I from the STMicroelectronics website. Then select the latest version compatible with your computer.

life.augmented	Pr	oducts 🗸 Search	Search	Contact Us
Products 👂 Applications	🖉 Solutions 🛛 🖌 Too	ols & Software 🛛 🏶 About ST	Sample & Buy	Support & Community 💄 I
		Get Software		
-	Part Number	▲ General Description 🖕 Latest	version Download 🗍 All versions	\$
	+ STM32CubeIDE-DEB	STM32CubeIDE Debian Linux Installer 1.7.0	Get latest Select vers	ion 🗸
	+ STM32CubeIDE-Lnx	STM32CubeIDE Generic Linux Installer 1.7.0	Get latest Select vers	ion 🗸
	+ STM32CubeIDE-Mac	STM32CubeIDE macOS Installer 1.7.0	Get latest Select vers	ion 🗸
_	+ STM32CubeIDE-RPM	STM32CubeIDE RPM Linux Installer 1.7.0	Get latest Select vers	ion 🗸
-	+ STM32CubeIDE-Win	STM32CubeIDE Windows Installer 1.7.0	Get latest Select vers	ion 🗸
-				

Figure 1: STM32CubeIDE Software Download

- 2. A license agreement from STMicroelectronics will show, and you must agree to log in using your STMicroelectronics account. Create an account if you do not have one.
- 3. After downloading, unzip the installer file and start the installation process. Just click next on the initial installation window and agree on the license agreement.



Figure 2: STM32CubeIDE Installation

STMicroelectronics STM32CubelDE –	×
icense Agreement	
Please review the license terms before installing STMicroelectronics STM32CubeIDE.	IDE
Press Page Down to see the rest of the agreement.	
TMicroelectronics Software License Agreement	^
SLA0048 Rev4/March 2018	
BY INSTALLING COPYING, DOWNLOADING, ACCESSING OR OTHERWISE USING THIS SOFTWARE PACKAGE OR ANY PART THEREOF (AND THE RELATED DOCUMENTATION) FROM STMICROELECTRONICS INTERNATIONAL N.V, SWISS BRANCH AND/OR ITS AFFILIATED COMPANIES (STMICROELECTRONICS), THE RECIPIENT, ON BEHALF OF HIMSELF OR HERSELF, OR ON BEHALF OF ANY ENTITY BY WHICH SUCH RECIPIENT IS EMPLOYED AND/OR ENGAGED AGREES TO BE BOUND BY THIS SOFTWARE PACKAGE LICENSE AGREEMENT.	*
If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install STMicroelectronics STM32CubeIDE.	
< Back I Agree Cano	el

Figure 3: STM32CubeIDE License Agreement

3. Next step is to determine the directory where you want the STM32CubeIDE software to be placed. You can just click next to use the default folder or you can select a different location.

DE STMicroelectronics STM32CubelDE	-		×
Choose Install Location			
Choose the folder in which to install STMicroelectronics STM32CubeIDE.			IDE
Setup will install STMicroelectronics STM32CubeIDE in the following different folder, dick Browse and select another folder. Click Next t	folder. To in o continue.	stall in a	
Destination Folder D:\Program Files Ext\ST\STM32CubeIDE\STM32CubeIDE_1.7.0	B <u>r</u> o	wse	
Space required: 2.3 GB Space available: 481.8 GB			
< <u>B</u> ack	<u>N</u> ext >	Ca	ncel

Figure 4: STM32CubeIDE Directory Location

4. Select optional components in the installation, like J-link and ST-Link drivers. It is highly recommended to include these drivers, which will be useful in debugging and uploading binary firmware files to the STM32WL chip.

hoose Components Choose which features of STM want to install.	icroelectronics STM32CubeIDE y	ou ID
Check the components you wa install. Click Install to start the	nt to install and uncheck the com installation.	nponents you don't want to
Select components to install:	SEGGER J-Link drivers	Description Position your mouse over a component to see its description.
Space required: 2.3 GB		

Figure 5: STM32CubeIDE Drivers

5. The progress bar will show as the installation begins.

STMicroelectronics STM32CubelDE			_		\times
Installing					IDE
Please wait while STMicroelectronics STM32Cut installed.	beIDE is being				IVE
Extract: org.edipse.launchbar.ui_2.4.100.202	012041937.jar.	100%			
Show details					
<u></u>					
	< <u>B</u> ack	<u>N</u> ext >		Car	ncel

Figure 6: STM32CubeIDE On-going Installation

6. During the installation, there are STMicroelectronics device drivers that will pop up. These are not needed by STM32WL, so you can leave them uninstalled.



Figure 7: STM32CubeIDE Optional Device Drivers

7. If there are no problems in the installation process, you should be finished and can now create a desktop shortcut for the STM32CubeIDE application.



Figure 8: STM32CubeIDE Installation Successful



Figure 9: STM32CubeIDE Installation Finished

8. Check if the installation of the STM32CubeIDE is successful by trying to run the app. It should have no errors. It will ask you for the workspace and you can leave the default location if you don't want to put it in another location. You also have the option to create multiple workspaces but only one should be active.

IDE STM32CubelDE Launcher		×
Select a directory as workspace		
STM32CubelDE uses the workspace directory to store its preferences and	development artifacts.	
Workspace: ⁹ C:\Users\RAK\STM32CubelDE\workspace_1.7.0	~	Browse
Use this as the default and do not ask again		
	Launch	Cancel

Figure 10: STM32CubeIDE Workspace Selection

orkspace_1.7.0 - STM32CubelDE	Carach Daviest Due Mindow Hale	- 0
Information Center 🔀	Search Project Kun Window Peip	
STM32CubelDE Home		
Start a project	Welcome to STM32CubeIDE	Support & Community
Fa	Welcome to of MiszouberDE	STM32
Start now	What's new	Twitter
STM32		Facebook
project		Youtube
	© Morres - C X tectorat bulkde Rauger ⊘r ⊙r 1	ST Home
		ST Community
L⊙	Manage 1 benchanner (Stranger 1 benchanner 1	ST Longevity Commitment
Start new project from	toolchains biotogeneric is used	
STM32CubeMX	<pre> concentrations, contraction contraction</pre>	
me (.ioc)		Standalone STM32 Too
	already installed	Cube
नि		O STUD2CubeMY
Import	•	STM32CubeMA
SW451M32 or TrueSTUDIO	How to access it : Right-click on the Project > Properties > Settings > Open Toolchain Manager	STM32CubeMon-Dwr
project		STM32CubeMon-RF
	Quick linke	STM32CubeMon-UCPD
	QUICK IIINS	STM32CubeProg
Import	Deed 0702000 helDE Decumentation	
STM32Cube example		
	Getting Started with STM32CubeIDE	Application Tools
		eDesianSuite
	W STM32 MPU wiki	
	**	ST-MC-Suite

Figure 11: STM32CubeIDE Application Running

RAK3172 on STM32CubeIDE with STM32WL SDK v1.0.0

STM32CubeIDE is a free IDE from STMicroelectronics, which you can use to develop firmware for various STM32 microcontrollers including the STM32WL series. It is a complete software IDE based on Eclipse where you can debug easily your code with built-in integration to tools like ST-Link and other features like STM32CubeMX. It is multiplatform software that can run on Windows, Linux, and macOS.

You cannot select RAK3172 directly on the STM32CubeIDE, but you can use the STM32WLE5CCU6 inside it with STM32WL SDK from STMicroelectronics to start your own custom firmware. This guide is only applicable to STM32WL SDK v1.0.0.

Getting STM32WL SDK v1.0.0

This guide only works on v1.0.0 of the SDK.

1. If you already have the STM32CubeIDE running on your machine, the next step is to download the STM32WL SDK v1.0.0 [□] from the STMicroelectronics website.

life.augmented	Products V Search		Search	Contact Us Er
🗰 Products 👂 Applications 🔗 Solution	ons 🦻 Tools & Software	About ST	Sample & Buy	Support & Community
STM32CubeWL ACTIVE				🜲 Save to MyST
STM32Cube M	CU Packag	ge for ST	M32WL se	eries (HAL,
Low-Layer API	s and CMS	, IS, File s	ystem, RT	OS, KMS,
Secure Engine	, Sub-GHz	Phy, LoR	aWAN an	d Sigfox
stacks - and ex	amples run	ining on S	ST boards)
Get Software	• Download databrief			
Overview Documentation Tools	& Software			

Figure 12: STM32WL SDK Download Page

		Products 🗸	Search	Search		Contact Us	English ∨
Products P Application	ons 🧷 Solution	ns 🖌 Tools & Software	STM32 Deve Zone	eloper 💮 About ST	Sample & Buy	Support & Community	💄 Login
	Part Number A STM32CubeWL	General Description STM32Cube MCU Pa	Get Softwar	Ce Supplier ≑ Download ST Get late Get from C	 ♦ All version Select v 1.2.0 ± 1.1.0 ± 1.0.0 ± 	ersion V	

Figure 13: STM32WL V1.0.0 Download

2. The downloaded files usually go to the download folder. You need to extract it then you will see the STM32WL SDK firmware folder.

> This PC > Hard Drive (D:) > Downloads > en.stm32cubewl_v1.0.0							
Name	Date modified	Туре	Size				
STM32Cube_FW_WL_V1.0.0	28/10/2020 4:38 PM	File folder					

Figure 14: STM32WL V1.0.0

3. The structure of the extracted files should be the same, as shown in **Figure 15**. You cannot just change this folder structure. This contains many examples related to the STM32WL chip.

> This PC > Hard Drive (D:) > Downloads > en.stm32cubewl_v1.0.0 > STM32Cube_FW_WL_V1.0.0					
Name ^	Date modified	Туре	Size		
_htmresc	28/10/2020 4:29 PM	File folder			
Documentation	28/10/2020 4:29 PM	File folder			
Drivers	28/10/2020 4:29 PM	File folder			
Middlewares	28/10/2020 4:29 PM	File folder			
Projects	28/10/2020 4:29 PM	File folder			
Utilities	28/10/2020 4:29 PM	File folder			
License.md	28/10/2020 4:29 PM	MD File	2 KB		
💼 package.xml	28/10/2020 4:29 PM	XML Document	1 KB		
README.md	28/10/2020 4:29 PM	MD File	2 KB		
Release_Notes.html	28/10/2020 4:29 PM	Firefox HTML Doc	16 KB		

Figure 15: STM32WL V1.0.0 Folder Structure

Modifications for the RAK3172

Files Modification Needed to Run STM32WL SDK LoRaWAN Examples to RAK3172

If you already have the STM32WL V1.0.0 SDK folder, there are only a few files that you need to update to be able to create firmware that will run on RAK3172.

NOTE

STM32 microcontroller firmware created using STM32CubeIDE (with the help of STM32CubeMX) have .ioc file. This is a configuration file created by the STM32CubeMX tool. This is a helpful tool in setting up peripherals and drivers quickly in the STM32 development ecosystem. However, once you do the file modification mentioned in this guide, you cannot create a new .ioc file or modify it via STM32CubeMX. Else, those modified files needed by the RAK3172 will be overwritten and will go back to their original state or the .ioc file.

In cases that you need to use STM32CubeMX to set up peripherals or drivers, you just need to do again the same modification as mentioned in this section.

1. Download the Low Level Development zip file ☐ from the RAK downloads site. Extract the zip file and inside the folder are four files that need to be copy-pasted on specific locations of the STM32WL V1.0.0 folder to make it compatible with RAK3172.

The majority of these files are for setting up the RF channel front end of the radio section on the STM32WL chip. Also, the startup file must be changed because the default startup on STM32WL SDK V1.0.0 is for the STM32WL55 series and not for STM32WLE5. The RAK3172 is based on STM32WLE5CCU6.

📝 NOTE

This guide will demonstrate how to run the **LoRaWAN_End_Node** example of the STM32WL SDK to RAK3172. If you need to run other LoRaWAN-related examples like **LoRaWAN_AT_Slave**, you need to update the files on that folder.

This PC > Hard Drive (D:) > Downloa	✓ [™] [™]		
Name	Date modified	Туре	Size
radio_board_if.c	12/08/2021 6:35 PM	C File	13 KB
🖻 radio_board_if.h	12/08/2021 6:35 PM	C/C++ Header	9 KB
🖻 radio_conf.h	12/08/2021 6:35 PM	C/C++ Header	5 KB
	12/08/2021 6:34 PM	Assembler Source	17 KB

Figure 16: RAK3172 Low Level Development Files

2. The radio related files radio_board_if.c , radio_board_if.h , and radio_conf must be placed in this location of the STM32WL SDK folder \STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN\Target . You have to overwrite or replace the old files.

Projects > NUCLEO-WL55JC > Applications > LoRaWAN > LoRaWAN_End_Node > LoRaWAN > Target							
Name	Date modified	Туре	Size				
🖻 lorawan_conf.h	16/06/2021 11:43 PM	C/C++ Header	5 KB				
🖻 mw_log_conf.h	16/06/2021 11:43 PM	C/C++ Header	3 KB				
🖹 radio_board_if.c	16/06/2021 11:43 PM	C File	9 KB				
🖻 radio_board_if.h	16/06/2021 11:43 PM	C/C++ Header	7 KB				
🖻 radio_conf.h	16/06/2021 11:43 PM	C/C++ Header	6 KB				
🖻 systime.h	16/06/2021 11:43 PM	C/C++ Header	2 KB				
🖻 timer.h	16/06/2021 11:43 PM	C/C++ Header	4 KB				

Figure 17: RAK3172 Radio Related Files for Modification



Figure 18: RAK3172 Radio Related Files Replaced

3. You also need to update the startup file. Place the startup_stm32wle5ccux.s file to this location
\STM32Cube_FW_WL_V1.0.0\Projects\NUCLE0-

WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\STM32CubeIDE\Application\Startup . There is a default startup file in that directory named startup_stm32wl55jcix.s and you need to delete that.

The updated startup folder should be the same, as shown in Figure 19.

« Applications > LoRaWAN > LoRaWAN_End_Node > STM32CubelDE > Application > Startup								
Name	Date modified	Туре	Size					
ា្នឹ startup_stm32wle5ccux.s	12/08/2021 6:34 PM	Assembler Source	17 KB					

Figure 19: RAK3172 Radio Related Files Replaced

Initial Build Test for the RAK3172 Custom Firmware

1. After doing the file modifications, the next step is to test if the LoRaWAN_End_Node example can be built without errors.

📝 NOTE

If this is your first time using STM32CubeIDE, it shows **Information Center** by default. Just close it and access the project on the left panel.

2. Open the STM32CubeIDE and click on File then Open Projects from File System .

Tour Tourse unseiter	Navigate Search Project Run Window	Heb	
New	Alt+Shift+N >		🏠 (> -> A* A*
Open File			
Open Projects from File Syst	tem		
Recent Files	>		
Close Editor	Ctrl+W	Welcome to STM22CubeIDE	Support & Community
Close All Editors	Ctrl+Shift+W	Welcome to STW32CuberDE	STM32
ave	Ctrl+S	What's new	
ave As			Twitter
ave All	Ctrl+Shift+S	STM32	Youtube
evert		cubalDe SWV enhancements :	PT Home
love			ST Community
ename	F2		ST Longevity Commitment
fresh	F5		
onvert Line Delimiters To		in microsov (122) m13 - UNUS MIC (2002) 000 Comp (2007) M20, upd. provide primo Data (2002) Control (2007) M20 (2002) M20	
	011.0	Endog 11 B Poliet Robow	
int	Ctri+P	 a 2 initialization (DCC 2014 and point of 1000 for 10000 for 10000 for 1000 for 1000 for 1000 for 1000 for 1000 for	Standalone STM32 T
nport		W HWC philips and Michael Machines 120 * prove that Anothesis is desired aswant is be described as is care W made endor/100000000 20 * [independentials is care file. J environe doub/J-0.320040000 20 * [independentials is care file.	tate -
(port		4 32 (MR (2) (MR COL server) 231 [232]	STM32CubeMX
operties	Alt+Enter		STM32CubeMonitor
vitch Workspace	>		STM32CubeMon-Pwr
start			STM32CubeMon-RF
it		Quick links	STM32CubeMon-UCPD
L#/			STM32CubeProg
Import STM32Cube		Read STM32CubeIDE Documentation	
example		—	
			Application Tools
		Getting Started with STM32CubeIDE	
			Consign Suite
		WZ STM32 MPLLwiki	
		W CHINZ III CHINA	Algobalder ST-MC-Suite
		W STM32 MCU wiki	
			Activate Windows

Figure 20: Open the Project in STM32CubeIDE

3. You then need to browse the project folder location by clicking the **Directory** button.

STM32CubeIDE Home		
Start a project	Import Projects from File System or Archive — — — X Import Projects from File System or Archive — — — X This wizard analyses the content of your folder or archive file to find projects and import them in the IDE. — — —	Support & Community
Start new STM32 project	Import source:	Twitter Facebook Youtube ST Home
Start new project from STM32CubeMX file (.ioc)	Folder Import as Deselect All	ST Community ST Longevity Commitment
Import sW45TM32 or True\$TUDIO project	Close newly imported projects upon completion Use installed project configuration to: Ø Search for nested projects Ø Detect and configure project natures Working sets	STM32CubeMX STM32CubeMonitor STM32CubeMonitor STM32CubeMon-Pwr STM32CubeMon-RF
Import STM32Cube example	Medd project to working sets New Working sets: Select Show other specialized import witards	STM32CubeMon-UCPD STM32CubeProg Application Tools
	O < Back Next > Finish Cancel	eDesignSuite AlgoBuilder ST-MC-Suite
	W/ STM32 MCI wiki	

Figure 21: Locate the Project Directory in STM32CubeIDE

4. You should locate this directory \STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-

WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node . Click on **STM32CubeIDE** folder once, then click the **Select Folder**.

IDE Browse for Folder				×
← → ~ ↑ 📙 « App	olications > LoRaWAN > LoRaWAN_End_	Node v 진	✓ Search LoRaV	/AN_End_Node
Organize 🔻 New folder	r			
This PC	Name	Date modified	Туре	Size
3D Objects	Core	16/06/2021 11:43 PM	File folder	
Desktop	EWARM	16/06/2021 11:43 PM	File folder	
Documents	LoRaWAN	16/06/2021 11:43 PM	File folder	
L Downloads	MDK-ARM	16/06/2021 11:43 PM	File folder	
h Music	STM32CubelDE	16/06/2021 11:43 PM	File folder	
Pictures				
Videos				
Local Disk (C:)				
Hard Drive (D:)				
A Network				
¥ -	<			>
Folder	STM32CubelDE			
			Select Folder	Cancel

Figure 22: Select the STM32CubeIDE Project Folder

5. You should now see the **STM32CubeIDE** checked and ready to be imported as **Eclipse project**. If not checked, click the checkbox and then the **Finish** button. It will take some time to fully import the whole project.

BAK[®] Documentation Center

DE Import Projec	ts from File System or Archive			×
Import Projects	from File System or Archive			5
This wizard analy	zes the content of your folder or archive file to find projects and import them in the IDE.			7
Import source:	D:\Downloads\STM32Cube_FW_WL_V1.1.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRa'	Di <u>r</u> ectory	<u>A</u> rchive	
type filter text		Sele	ct All	
Folder	Import as	Desel	ect All	
STM32C	ubelDE Eclipse project	<u>_</u>		
		1 of 1 selected		
	monted projects upon completion	[] [] Hide already	/ open proje	cts
Use installed pro	ject configurators to:			
Search for ne	sted projects			
Detect and <u>c</u>	onfigure project natures			
Working sets				
Add proje	t to working sets		Ne <u>w</u>	1
Working sets:		~	S <u>e</u> lect	
		Show other specialized	l import wiza	ards
?	< <u>B</u> ack <u>N</u> ext >	<u>F</u> inish	Cancel	

Figure 23: Open the LoRaWAN_End_Node Project

 After the successful import, you should now see the LoRaWAN_End_Node project structure on the left side of the STM32CubeIDE.

workspace_1.7.0 - STM32CubelDE									-	۵	×
<u>File Edit Source Ketactor Navigate Search Project Kun Wi</u>	indow <u>H</u> elp • O • O • O • I № I № I № I № I I I I I I I I I I I	• 图 • tb c	* \(\- \) + \ = +	A						Q i 😭	E Cha
Project Explorer X							- 0	₽ Outline 🛛 @	Build Targets		
 ChawWANE_End_Node (in STM32CubeIDE) Sig) Includes 								There is no active e	ditor that provides	: an outline.	
	🖹 Problems 🛛 🧔 Tasks 📮 Console 🔲 Properti	es		78-0	Build Analyzer	🛛 🚊 Static Stat	k Analyzer			28	' 🗆
	Description	Resource	Path	Location	Manage Back						
					Region	Memory Details	End address	Size	Eree	Head	
						ctare address	and ordereds	- tax		0.100	
							-	Activate Win	dows		_
					<		(Go to Settings to	activate Wind	ows.	>
00E LoRaWAN_End_Node	,			,							

Figure 24: Open the STM32CubeIDE Project

7. With the modified files already implemented, you can check if the files are updated by checking the startup file startup_stm32wle5ccux.s and the radio_board_if.c . The startup file must be updated and show startup_stm32wle5ccux.s . You should see #if defined(RAK3172_RF_CHANNEL_SWITCH) in line 72 of radio_board_if.c file, as shown in Figure 25. If not, then you are not successful in changing these files with the Low Level Development required modification.

workspace 1.7.0 - LoRaWAN End Node/Application/Target/radio board if.c - STM32CubeIDE			- 0	×
File Edit Source Refactor Navigate Search Project Run Window Help				
, , , , , , , , , , , , , , , , , , ,	10		Q : 😭	
		■ Outline 😚 🛞 Bu	uild Targets	
	<u> </u>		a 🔊 🔊 a	3 # 8
<pre>> Cressing Application > Application ></pre>		 radio_board_it RBI_Init(void) RBI_OcnfigR: RBI_GetTxCon RBI_GetTxCon RBI_STCXO(vc RBI_ISDCDC(vc 	z v ih :int32_t d):int32_t Switch(RBLSw fig(void):int3 JpTime(void) sid):int32_t oid):int32_t	vitch_Typel 32_t : int32_t
60 Byze_inte_ser accale speed = a 10 st ebb, not ven intan;	> <			>
Problems @ Tasks Console 22 Properties	Build Analyze	r 🕅 🛋 Static Stack	Analyzer	
CDT Build Console [LoRaWAN_End_Node]	Memory Region	5 Memory Details	,	28
	Region	Start address	End address	s Si
	<			>
/LoRaWAN_End_Node/Application/Target/radio_board_if.c				

Figure 25: Open the STM32CubeIDE Project

8. You can now try to build the project by setting up the build configuration to release so that a .bin file will be generated.

VOTE

If you have an ST-LINK debugging tool, you can also choose **Debug** instead of **Release**.

Inc	- (Analisation Tarant (adia based 26 - STB 4300-ball)			~
File Edit Source Refactor Navigate	Search Project Run Window Help		- 0	^
Int guilt goulde included in <u>H</u> ungale Int will be a state of the state	0 jeft Frider Han Hindow Help () 1 個 Open Project + 2 タ マーム 図 目 町 1 日 マ 行 マ ひ マ ロ マ 一	0	Q 😭 🖪	to MX
Project Explorer 💥 📃 🗆	c radie Close Project		📴 Outline 🔀 🛞 Build Targets 📑	- 0
Image: Second Secon	Build All Ctrl+B Build Configurations Set Active Build Configurations Manage Build Project Manage Build Working Set Set Active Build Automatically Build Project Build Automatically Set Active by Working Set Build Targets Set Active by Working Set C/C++ Index Inter = {0}; C/C++ Index Inter = {0}; C/C++ Index Inter = {0}; Set Active by Dorking Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Build Targets Inter = {0}; C/C++ Index Inter = {0}; Set Active by Dorking Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set Set Active by Working Set		<pre>I^A_Z N N N N N N N N N N N N N N N N N N N</pre>	h_Type : t32_t
	80 gpio_init_structure.Speed = GPIO_SPEED_FREQ_VERY_HIGH; * * Problems @ Tasks © Console 23 Problems @ Tasks © Console 23 * • *	> Build Analy	 zer ☆ ≝ Static Stack Analyzer □ 	,
	CDI Build Console [LoKaWAN_End_Node]			
		Memory Regi	ons Memory Details	C;
		<).
< >>				
LoRaWAN_End_Node/Application/Targ	et/radio_board_if.c			

Figure 26: Configure Build to Release

BAK[®] Documentation Center

workspace_1.7.0 - LoRaWAN_End_Node	e/Applicat	ion/Targ	jet/radio_board_if.c	- STM32CubelD	E				- 0	×
File Edit Source Refactor Navigate	Search	Project	Run Window	Help						
📑 = 🔚 🐘 = 🗞 = 🐘 🔍 [0	@ : 😭	Op	pen Project			0			Q i 😭	
Project Explorer 22	R radie	Cle	ose Project			-	E	Dutline 😚 🛞 Bi	uild Targets	
	61	Bu Bu	uild All	Ctrl+B			^		a, 🔊 🔊 e	# 8
V DE LoRaWAN_End_Node (in STM32Cub	62	Bu	uild Configurations	>				radio_board_	f.h	
> 🔊 Includes	63	Bu	uild Project		dd the correspondent files in the IDE workspace */			 RBI_Init(void) 	: int32_t	
Application	65	Bu	uld Working Set	>	the preprocessor definitions or in platform.h */			 RBI_Delnit(vo RBI_ConfigRE 	id) : int32_t Switch/PPL Su	itch Tunol
> 🗁 App	67 -	Cle	ean					 RBI GetTxCor 	fiq(void) : int3	32 t
V 👝 Startup	68	Bu	uild Automatically		Init() but not supported by MX*/			RBI_GetWake	UpTime(void)	: int32_t
S startup_stm32wle5ccux.s	70	Bu	uild Targets	>	<pre>icture = {0};</pre>			RBI_IsTCXO(v RBI_IsTCXO(v)	oid) : int32_t	
Iarget Iarget Iarget difference	71	0	(C++ Index	、 、	тсн)			RBI_ISDCDC(\	roid) : int32_t	
> 🗁 Drivers	73	Ge	enerate Report		ack */					
> 👝 Middlewares	74	dh Ge	enerate Code							
> Utilities	76				pin */					
STM32WL55JCIX_FLASH.Id	78	Bbr	openies 20_1011_Structur	re.noue = u	<pre></pre>					
	79 80	gpi	.o_init_structur	re.Pull = G	PIO_NOPULL; PIO_SPEED_EREO_VERY_HIGH·					
		<		crispeca - a		>	× <			>
	Prob	lems 🎜	Tasks 📄 Conso	le 🕸 🔲 Pror	perties	🗈 Build	d Analyzer	😤 🛋 Static Stac	k Analyzer	
						019		_		- 8
	CDT Build	l Console	e [LoRaWAN_End_N	lode]						
					^	Memo	ory Regions	Memory Details		
						Regio	n	Start address	End address	Si
						inegro				
					×	<				>
< >> (LoPaWAN End Node/Application (Trans	<	oard if -	-		>					
	ler/ladio_p	ioaru_if.c	-				:			

Figure 27: Build the Project

<pre>workspace_17.2 - LoRAWAL_Ind_Node/Application/Engrafies_SIMA2CABDE</pre>	_						
<pre>Elle Edi Suore Refect Mavigne Seach Project Bon Window Hep Project Egiorer St Content St Cont</pre>	workspace_1.7.0 - LoRaWAN_End_Node	/Application/Target/radio_board_if.c - STM32CubelDE			- 0	×	
• • • • • • • • • • • • • • • • • • •	<u>File Edit Source Refactor Navigate</u>	Se <u>a</u> rch <u>P</u> roject <u>R</u> un <u>W</u> indow <u>H</u> elp					
Project Explore: 21 Project Explore: 2	📑 🕶 🔚 🔞 😻 🕶 🗞 🕶 🔜 🔌 🛙	🎭 🖥 🗙 🚳 🕶 🙆 🕶 🚱 🕶 🔅 🐐 🔹 🜑 🖛 🎝 👘 🔔 🖋 🕶 🖉 🖉 🖉 🖉 👘 👘 👘 🖓 🕶 🖓 🕶 👘 👘 👘	0		Q 🗄 😭 🛛	Ec MX	
<pre> Construction Construction</pre>	陷 Project Explorer 🐹 👘 🗖	🖻 radio_board_if.c 😒		🗄 Outline 🛛 🛞 B	Juild Targets		
<pre> v I LAWAW End Node (in STMS2Ce) v I find define (USE_55*_DRTWR) v Applexin v Apple</pre>	E 🕏 🍸 🕴	61 /* USER CODE END RBI_Init_1 */	^	E	l ^a z 😿 🗙 🖌	# 8	
CDT Build Console [LoRaWAN_End_Node] arm-none-eabi-size LoRaWAN_End_Node.elf arm-none-eabi-size LoRaWAN_E	 ✓ IL LORAWAN_End_Node (in STM32Cub) > ∰ Einaries > ∰ Includes > Application > Apple > ⊕ App > ⊕ Startup > ⊕ Target > ⊕ Middlewares > ⊕ Middlewares > ⊕ Utilities ■ LoRaWAN_End_Node.ioc ➡ STM32WLSSICIX_FLASH.Id 	<pre>1 1 /* USER CODE END RBI_Init 1 */</pre>					
Problems Tasks Console Con		<pre>co gpio_init_structure.speeu = drid_speeu = keQ_vekY_hidh;</pre>	~	<	_	>	
COT Build Console [LoRaWAN_End_Node] arm-none-eabi-size LoRaWAN_End_Node.elf arm-none-eabi-objcopy - 0 binary LoRaWAN_End_Node.elf * LoRaWAN_End_Node.list" arm-none-eabi-objcopy - 0 binary LoRaWAN_End_Node.elf * LoRaWAN_End_Node.list" text data bss dec hex filename 62880 384 8368 71632 11704 LoRaWAN_End_Node.elf Finished building: LoRaWAN_End_Node.list 18:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Comparison of the state of the st						-	
CDT Build Console [LoRaWAN_End_Node] arm-none-eabi-size LoRaWAN_End_Node.elf arm-none-eabi-objdump -h -5 LoRaWAN_End_Node.elf > "LoRaWAN_End_Node.list" arm-none-eabi-objdump -h -5 LoRaWAN_End_Node.elf > "LoRaWAN_End_Node.list" arm-none-eabi-objcopy -0 binary LoRaWAN_End_Node.elf * LoRaWAN_End_Node.elf * Ea888 384 8368 71632 117d0 LoRaWAN_End_Node.elf Einished building: default.size.stdout Finished building: LoRaWAN_End_Node.list 18:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Curror of the finished hold hold hold hold hold hold hold hol			Dulid Anal	yzer 23 <u>m</u> static stat	.k Analyzer		
Arm-none-eabi-objcopy - 0 binary LoßalAM_End_Node.elf "LoßalAM_End_Node.bin" text data bss dec hex filename 62880 384 8368 71632 117d0 LoßalAM_End_Node.elf Finished building: default.size.stdout Finished building: LoßalAM_End_Node.bin Finished building: LoßalAM_End_Node.list 18:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) I DBAWAN End Node	ſ	CDT Build Console [LoRaWAN_End_Node]	LoRaWAN_E /LoRaWAN_ 6:08:03 PM	nd_Node.elf - End_Node/Release -	Aug 13, 2021	•	
62880 384 8368 71632 117d0 LoRaWAN_End_Node.elf Finished building: default.size.stdout Image: Construction of the co		arm-none-eabi-objcopy -0 binary LoRaWAN_End_Node.eif "LoRaWAN_End_Node.bin" text data bss dec hex filename	Memory Reg	gions Memory Details			
Finished building: derault.sile.stdout Finished building: LoRaWAN_End_Node.bin Finished building: LoRaWAN_End_Node.list 18:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms)		62880 384 8368 71632 117d0 LoRaWAN_End_Node.elf	Region	Start address	End address	Si	
Finished building: LoRawAN_End_Node.bin Image: RAM1 0x2000000 0x2000000 32 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 0x2000000 0x2001000 32 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 0x2000000 0x2001000 32 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 0x2001000 32 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 0x2001000 32 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 0x2001000 32 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 Image: RAM2 Image: RAM2 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 Image: RAM2 Image: RAM2 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 Image: RAM2 Image: RAM2 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 Image: RAM2 Image: RAM2 Is:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) Image: RAM2 Image: RAM2 Image: RAM2		Finished building: default.size.stdout	m ROM	0x0800000	0x08040000	25	
Finished building: LoRaWAN_End_Node.list 18:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) *		Finished building: LoRaWAN_End_Node.bin	ERAM1	0x20000000	0x20008000	32	
18:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms) IbBaWAN End Node		Finished building: LoRaWAN_End_Node.list	RAM2	0x20008000	0x20010000	32	
< > < > < > < > < > < > < > < > < > < >		18:08:04 Build Finished. 0 errors, 1 warnings. (took 9s.185ms)					
I JBRAWAN End Node	<	< v	۲.			>	
	VE LoRaWAN End Node		1				

Figure 28: Successful Project Build

RAK3172 on STM32CubeIDE with STM32WL SDK v1.2.0

The previous guide is for STM32WL SDK version 1.0.0. This guide is compatible with STM32WL SDK v1.2.0.

Getting STM32WL SDK v1.2.0

1. If you already have the STM32CubeIDE running on your machine, the next step is to download the STM32WL SDK v1.2.0 ^I from the STMicroelectronics website.



Figure 29: STM32WL SDK Download Page

	ed	Products ~	Search	Search	Contact Us	English 🗸
III Products 👂 Appl	lications 🧷 S	olutions 🦻 Tools & Software	STM32 Developer Zone	⊕ About Sar ST Buy	mple & Support & y Community	💄 Login
-	Part Numbe + STM32Cub	eWL STM32Cube MCU Pack	Get Software	Download Get latest Get from GitHub	All versions ♦ Select version ∨ 1.2.0 ★ ↓ 1.1.0 ★ ↓ 1.0.0 ★ ↓	

Figure 30: STM32WL V1.2.0 Download

 The downloaded files usually go to the download folder. You need to extract it then you will see the STM32WL SDK firmware folder.

en.stm32cubewl_v1-	2-0			>
🕀 New -	0 6 0] 🖻 🗓 ↑↓ Sort ~	\equiv View \sim	
\leftarrow \rightarrow \checkmark \uparrow	📒 « Downloads >	en.stm32cubewl_v1-2-0 >	✓ C	bewl_v1-2-0
✓		Name	Date modified Typ	e Size
E Desktop	*	STM32Cube_FW_WL_V1.2.0	9/25/2022 11:50 PM File	folder
🛓 Downloads	*	1		
Documents	*			

Figure 31: STM32WL V1.2.0

3. The structure of the extracted files should be the same, as shown in **Figure 32**. You cannot just change this folder structure. This contains many examples related to the STM32WL chip.

STM32Cube_FW_WL_V	/1.2.0			- 0 ×
🕂 New - 🏑		↑↓ Sort ~ 🗮 View ~		
$\leftarrow \rightarrow \checkmark \uparrow$	🚞 « work > STM32Cube_FW_W	CCSearch STI	M32Cube_FW_WL_V1.2.0	
声 Blink_Working	Name	Date modified	Туре	Size
> 👼 WisBlock-Aut	_htmresc	8/16/2022 11:23 AM	File folder	
) 🔿 OpeDrive - Peri	Documentation	8/16/2022 11:15 AM	File folder	
	Drivers	8/16/2022 11:19 AM	File folder	
🗸 🛄 This PC	Middlewares	8/16/2022 11:20 AM	File folder	
> 📒 Desktop	📒 Projects	8/16/2022 11:22 AM	File folder	
> 📑 Documents	📒 Utilities	8/16/2022 11:23 AM	File folder	
> 🛓 Downloads	D package.xml	2/3/2022 2:55 PM	XML Document	1 KB
> 🕖 Music	🦁 Package_license.html	2/3/2022 2:55 PM	Brave HTML Docu	82 KB
> 🔀 Pictures	Package_license.md	2/3/2022 2:55 PM	Markdown Source	12 KB
> 🚺 Videos	README.md	2/3/2022 2:55 PM	Markdown Source	2 KB
> 💾 OS (C:)	🦁 Release_Notes.html	2/3/2022 2:55 PM	Brave HTML Docu	35 KB
> 🛬 Network				

Figure 32: STM32WL V1.2.0 Folder Structure

Modifications for the RAK3172

Files Modification Needed to Run STM32WL SDK LoRaWAN Examples to RAK3172

If you already have the STM32WL V1.2.0 SDK folder, there are only a few files that you need to update to be able to create firmware that will run on RAK3172.

📝 NOTE

STM32 microcontroller firmware created using STM32CubeIDE (with the help of STM32CubeMX) have .ioc file. This is a configuration file created by the STM32CubeMX tool. This is a helpful tool in setting up peripherals and drivers quickly in the STM32 development ecosystem. However, once you do the file modification mentioned in this guide, you cannot create a new .ioc file or modify it via STM32CubeMX. Else, those modified files needed by the RAK3172 will be overwritten and will go back to their original state or the .ioc file.

In cases that you need to use STM32CubeMX to set up peripherals or drivers, you just need to do again the same modification as mentioned in this section.

 Download the Low Level Development zip file for v1.2.0 ^I from the RAK downloads site. Extract the zip file and inside the folder are four files that need to be copy-pasted on specific locations of the STM32WL V1.2.0 folder to make it compatible with RAK3172.

The majority of these files are for setting up the RF channel front end of the radio section on the STM32WL chip. Also, the startup file must be changed because the default startup on STM32WL SDK V1.2.0 is for the STM32WL55 series and not for STM32WLE5. The RAK3172 is based on STM32WLE5CCU6.

📝 NOTE

This guide will demonstrate how to run the **LoRaWAN_End_Node** example of the STM32WL SDK to RAK3172. If you need to run other LoRaWAN-related examples like **LoRaWAN_AT_Slave**, you need to update the files on that folder.

This PC > Hard Drive (D:) > Downloa	ds > RAK3172_Low_Level_Develo	pment	v ت	3
Name	Date modified	Туре	Size	
radio_board_if.c	12/08/2021 6:35 PM	C File		13 KB
🖻 radio_board_if.h	12/08/2021 6:35 PM	C/C++ Header		9 KB
🖻 radio_conf.h	12/08/2021 6:35 PM	C/C++ Header		5 KB
₽a startup_stm32wle5ccux.s	12/08/2021 6:34 PM	Assembler Source		17 KB

Figure 33: RAK3172 Low Level Development Files

2. The radio related files radio_board_if.c , radio_board_if.h , and radio_conf must be placed in this location of the STM32WL SDK folder \STM32Cube_FW_WL_V1.2.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN\Target . You have to overwrite or replace the old files.

« Projects > NUCLEO-WL55JC > Applications > LoRaWAN > LoRaWAN_End_Node > LoRaWAN > Target				
Name	Date modified	Туре	Size	
🖹 lorawan_conf.h	16/06/2021 11:43 PM	C/C++ Header	5 KB	
🗈 mw_log_conf.h	16/06/2021 11:43 PM	C/C++ Header	3 KB	
📄 radio_board_if.c	16/06/2021 11:43 PM	C File	9 KB	
🖻 radio_board_if.h	16/06/2021 11:43 PM	C/C++ Header	7 KB	
🗈 radio_conf.h	16/06/2021 11:43 PM	C/C++ Header	6 KB	
🖻 systime.h	16/06/2021 11:43 PM	C/C++ Header	2 KB	
🖻 timer.h	16/06/2021 11:43 PM	C/C++ Header	4 KB	

Figure 34: RAK3172 Radio Related Files for Modification

Replace or Skip Files —	×
Moving 3 items from RAK3172_Low_Level_Development to Target	
The destination has 3 files with the same names	
\checkmark Replace the files in the destination	
Skip these files	
🖒 Let me decide for each file	
Fewer details	

Figure 35: RAK3172 Radio Related Files Replaced

3. You also need to update the startup file. Place the startup_stm32wle5ccux.s file to this location
\STM32Cube_FW_WL_V1.2.0\Projects\NUCLEO-

WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\STM32CubeIDE\Application\User\Startup . There is a default startup file in that directory named startup_stm32wl55jcix.s and you need to delete that.

The updated startup folder should be the same, as shown in Figure 36.

🔜 « Applications » LoRaWAN » LoRaWAN_End_Node » STM32CubelDE » Application » Startup					
Name	Date modified	Туре	Size		
ជា startup_stm32wle5ccux.s	12/08/2021 6:34 PM	Assembler Source	17 KB		

Figure 36: RAK3172 Radio Related Files Replaced

- 4. After the modifications above, there are minor changes needed to be adjusted on other source files.
- On \STM32Cube_FW_WL_V1.2.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\Core\Inc\platform.h, you need to comment out #define USE_BSP_DRIVER. It should be //#define USE_BSP_DRIVER.
- On \STM32Cube_FW_WL_V1.2.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN\Target\lorawan_conf.h, you need to change the version of LoRaWAN to 1.0.3 which is the commonly used LNS version at the moment of this writing. To do this, you have to change LORAMAC_SPECIFICATION_VERSION to 0x01000300. It should look like this

#define LORAMAC_SPECIFICATION_VERSION 0x01000300 . However, if you are using LoRaWAN version 1.0.4 on your LNS, you do not need to perform this step since the DevNonce will be handled correctly.

Initial Build Test for the RAK3172 Custom Firmware

1. After doing the file modifications, the next step is to test if the LoRaWAN_End_Node example can be built without errors.

📝 NOTE

If this is your first time using STM32CubeIDE, it shows **Information Center** by default. Just close it and access the project on the left panel.

2. Open the STM32CubeIDE and click on File then Open Projects from File System .



Figure 37: Open the Project in STM32CubeIDE

3. You then need to browse the project folder location by clicking the **Directory** button.

IDE workspace_1.7.0 - STM32CubelDE <u>File</u> <u>E</u> dit <u>S</u> ource Refac <u>tor N</u> avigate Se <u>a</u> rch	Projest <u>B</u> un <u>W</u> indow <u>H</u> elp	– 0 X
Information Center STM32CubeIDE Home		👌 🗘 🖒 🖈 🕂 🖻 🗗
Start a project	Import Projects from File System or Archive X Import Projects from File System or Archive This wizard analyzes the content of your folder or archive file to find projects and import them in the IDE. Import source I Directory Archive	Support & Community Twitter Facebook Voutube
Start new project from STM32CubeMX file (Joc)	type filter text Select All Folder Import as Deselect All 0 of 0 selected Hide already open projects	ST Home ST Community ST Longevity Commitment
EP swististista trusstudio project	Close newy imported projects upon competition Use installed project configured projects Show other specialized import witards	STM32CubeMX STM32CubeMonFor STM32CubeMon-For STM32CubeMon-RF STM32CubeMon-UCPD STM32CubeMon-UCPD STM32CubeMon-UCPD
	⑦ < Back Next> Finish Cancel W STM32 MCU wiki ■ Evaluere Minetice Nexus in STM22CubalTPE	Application Tools Constraints AppBuilder AppBuilder Activate Windows Go to Settings to activate Windows.
Type here to search	o # \$,# # • • • • # # # # # # # #	26°C Haze へ 🕼 🦟 🕬 🗰 🦧 ENG 10:39 PM 🌄

Figure 38: Locate the Project Directory in STM32CubeIDE

4. You should locate this directory \STM32Cube_FW_WL_V1.2.0\Projects\NUCLEO-

WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node . Click on **STM32CubeIDE** folder once, then click the **Select Folder**.

DE Browse for Folder				×
← → • ↑ <mark> </mark> « /	Applications > LoRaWAN >	LoRaWAN_End_Node v	Ö 🔎 Search Lo	aWAN_End_Node
Organize 🔻 New fo	lder			· · · · · · · · · · · · · · · · · · ·
This PC	Name	Date modified	Туре	Size
3D Objects	Core	16/06/2021 11:43	PM File folder	
Desktop	EWARM	16/06/2021 11:43	PM File folder	
Documents	LoRaWAN	16/06/2021 11:43	PM File folder	
Downloads	MDK-ARM	16/06/2021 11:43	PM File folder	
b Music	STM32CubelDE	16/06/2021 11:43	PM File folder	
Pictures				
Videos				
🏪 Local Disk (C:)				
Hard Drive (D:)				
A Network				
- · · · · · · · · · · · · · · · · · · ·	/ <			>
Fol	der: STM32CubelDE			
			Select Folder	Cancel

Figure 39: Select the STM32CubeIDE Project Folder

5. You should now see the **STM32CubeIDE** checked and ready to be imported as **Eclipse project**. If not checked, click the checkbox and then the **Finish** button. It will take some time to fully import the whole project.

INF Import Projects from File System or Archive		— 🗆 X			
Import Projects from File System or Archive This wizard analyzes the content of your folder or archive file to find projects and import them in the IDE.					
Import source: D:\Downloads\STM32Cube_FW_WL_V1.1.0\Projects\NUCLEO-WL55J0	C\Applications\LoRaWAN\LoRa	✓ Directory <u>A</u> rchive			
type filter text		<u>S</u> elect All			
Folder	Import as	Deselect All			
Close newly imported projects upon completion Use installed project configurators to: Search for nested projects Detect and configure project natures Working sets Add project to working sets	cupse project	1 of 1 selected ☐ <u>H</u> ide already open projects Ne <u>w</u>			
Wgrking sets:		Select Show other specialized import wizards			
?	< <u>B</u> ack <u>N</u> ext >	<u>F</u> inish Cancel			

Figure 40: Open the LoRaWAN_End_Node Project

 After the successful import, you should now see the LoRaWAN_End_Node project structure on the left side of the STM32CubeIDE.

_											
workspace_1.7.0 - STM32CubelDE	deve I lete								-	٥	×
File Edit Source Keractor Navigate Search Project Kun Win		- 20 - ***		8						0 :	
		• 81 • 4 4		U						× : t	
In Project Explorer 32 P 45 Y 8 P C LoRAWAR, End Node (in STM32CubeID) > 20 Includes > 20 Application > 20 Doc > 20 Doc > 20 Middlewares > Widdlewares > Widdleware								E Outline 23 (@) Build Targets	: an outlin	<u>в</u> .
	🖹 Problems 🛛 🧔 Tasks 📮 Console 🔲 Propertie	es		78-0	Build Analyzer	🕴 🚊 Static Stac	k Analyzer			📑 8	
	0 items		14.4								
	Description	Resource	Path	Location	Memory Regions	Memory Details					
					Region	Start address	End address	Size	Free	Used	
								Activate Win	dows		
					<		(Go to Settings to	activate Wind	OWS.	>
LoRaWAN_End_Node	\$			>							

Figure 41: Open the STM32CubeIDE Project

7. With the modified files already implemented, you can check if the files are updated by checking the startup file startup_stm32wle5ccux.s and the radio_board_if.c . The startup file must be updated and show startup_stm32wle5ccux.s . You should see #if defined(RAK3172_RF_CHANNEL_SWITCH) in line 72 of radio_board_if.c file, as shown in Figure 42. If not, then you are not successful in changing these files with the Low Level Development required modification.

workspace_1.7.0 - LoRaWAN_End_Node	/Application/Target/radio_board_if.c - STM32CubelDE			- 0	×
<u>File Edit Source Refactor N</u> avigate	Se <u>a</u> rch <u>P</u> roject <u>R</u> un <u>W</u> indow <u>H</u> elp				
📑 🕶 🔚 🐚 🛞 🕶 🗞 🕶 🔜 ! 🔌 ! i	9월 📸 ▾ 🚳 ▾ 🖸 ▾ 🞯 ▾ 🛊 ▾ 🔕 ▾ 💁 🖋 ▾ 💷 🖗 🗐 👖 월 ૫ 종종 ▾ 🖓 ▾ ♥> 라 🔶 ▾ 리 ▾ 🏻 🛃	0		Q 🔡	Ec MX
Project Explorer 💥 📃 🗖	ic) radio_board_if.c ⊠		Outline 🔀 💿 Bui	ild Targets	- 8
E 🔄 🍸 🕴	61 /* USER CODE END RBI_Init_1 */	^	⊟ ↓ª	z 😿 🗙 🔹	# 8
 ✓ IL LORAWAN_End_Node (in STM32Cub) > Di Includes > △ Application > △ App > ○ Core > △ Startup > △ Target > △ radio_board_if.c > △ Middlewares > ◯ Utilities I LoRaWAN_End_Node.ioc ⊇ STM32WLSSJCIX_FLASH.Id 	<pre>62 #if defined(USE_BSP_DRTVER) // * code generated by MX does not support BSP */ 63 /* code generated by MX does not support BSP */ 64 /* In order to use BSP_driver, add the correspondent files in the IDE workspace */ 65 /* and define USE_BSP_DRTVER in the preprocessor definitions or in platform.h */ 66 return BSP_RADID_Init(); 67 #elif defined(MX_NUCLEO_MLSSIC1) 68 /* should be calling BSP_RADID_Init() but not supported by MX*/ 69 70 GPIO_InitTypeDef gpio_init_structure = {0}; 71 72 #if defined(RAS1372_RF_CHANNE_SNITCH) 73 /* table the Radio Switch Llock */ 74 RF_SM_CTRL1_GPIO_CLK_ENABLE(); 75 RF_SM_CTRL2_GPIO_CLK_ENABLE(); 76 /* Configure the Radio Switch pin */ 77 gpio_init_structure.Pin = RF_SM_CTRL1_PIN; 78 gpio_init_structure.Pull = GPIO_NOPULL; 79 gpio_init_structure.Pull = GPIO_NOPULL; 79 gpio_init_structure.Pull = GPIO_NOPULL; 79 gpio_init_structure.Pull = GPIO_NOPULL; 79 gpio_init_structure.Pull = GPIO_NOPULL; 70 gpio_init_structure.Pull = GPIO_NOPULL; 71 gpio_init_structure.Pull = GPIO_NOPULL; 72 gpio_init_structure.Pull = GPIO_NOPULL; 73 gpio_init_structure.Pull = GPIO_NOPULL; 74 gpio_init_structure.Pull = GPIO_NOPULL; 75 gpio_init_structure.Pull = GPIO_NOPULL; 76 gpio_init_structure.Pull = GPIO_NOPULL; 77 gpio_init_structure.Pull = GPIO_NOPULL; 78 gpio_init_structure.Pull = GPIO_NOPULL; 79 gpio_init_structure.Pull = GPIO_NOPULL; 79 gpio_init_structure.Pull = GPIO_NOPULL; 70 gpio_init_structure.Pull = GPIO_NOPULL; 70 gpio_init_structure.Pull = GPIO_NOPULL; 71 gpio_init_structure.Pull = GPIO_NOPULL; 72 gpio_init_structure.Pull = GPIO_NOPULL; 73 gpio_init_structure.Pull = GPIO_NOPULL; 74 gpio_init_structure.Pull = GPIO_NOPULL; 75 gpio_init_structure.Pull = GPIO_NOPULL; 76 gpio_init_structure.Pull = GPIO_NOPULL; 77 gpio_init_structure.Pull = GPIO_NOPULL; 78 gpio_init_structure.Pull = GPIO_NOPULL; 79 gpio_init_structure.Pull = GPIO_NOPULL; 70 gpio_init_structure.Pull = GPIO_NOPULL; 71 gpio_init_structure.Pull = GPIO_NOPULL; 72 gpio_init_structure.Pull = GPIO_NOPUL; 73 gpio_init_structure.Pull = GPI</pre>		radio_board_if. RBL_init(void) : RBL_Oenit(void) : RBL_Oenit(void) : RBL_GetWaceJ RBL_GetWaceJ RBL_GetWaceJ RBL_SCDC(vc	h int32_t i): int32_t witch(RBI_Swi ig(void): int33 pTime(void): id): int32_t yid): int32_t	itch_Typel 2_t int32_t
	<	>			>
	🔋 Problems 🔎 Tasks 😑 Console 🖄 🥅 Properties 👘 🗖	Build Analyzer	🖂 🛋 Static Stack	Analyzer	
	X ↓	Memory Regions	Memory Ostails		28
		Region	Start address	End address	Ci.
< >	()	<			>
C /LoRaWAN_End_Node/Application/Targ	et/radio_board_if.c				

Figure 42: Open the STM32CubeIDE Project

8. The next step is to ensure that a bin file will be generated on your release build. Go to the properties and ensure that *Convert to binary file (-O binary)* is checked.

Properties for LoRaWAN_En	nd_Node	
type filter text	Settings	
 > Resource Builders > C/C++ Build Build Variables Environment Logging > Settings > C/C++ General CMSIS-SVD Settings Project References Run/Debug Settings 	Configuration: Release Tool Settings Build Steps Build Artifact Binary Parse MCU Toolchain Convert to binary file (-O binary MCU Settings Convert to Intel Hex file (-O iner MCU Settings Convert to Motorola S-record file MCU GCC Assembler Convert to Verilog file (-O verilog Bebugging Show size information about build Preprocessor Show size information about build	rs Error Parsers) x) le (-O srec) g) symbols) file (-O symbolsrec) iilt artifact
	 ➢ Miscellaneous ✓ S MCU GCC Compiler △ General 	

Figure 43: Bin generation settings

9. You can now try to build the project by setting up the build configuration to release so that a .bin file will be generated.

NOTE

If you have an ST-LINK debugging tool, you can also choose **Debug** instead of **Release**.

workspace_1.7.0 - LoRaWAN_End_Node	/Application/Target/radio_board_if.c - STM32CubeIDE		- 0	×
<u>File Edit Source Refactor Navigate</u>	Search Project Run Window Help			
	9) 🗄 📸 👘 Open Project 👘 🛃 💌 📴 🖉 τ 🔛 🕼 🔟 👖 🗄 😓 τ 🏷 τ [*] 🗘 τ 🖓 τ τ το τ Γ	1 0	Q 🛛 😭	
🍋 Project Explorer 🐹 👘 🗖	c radie Close Project		📴 Outline 🙁 💿 Build Targets	
 ✓ ELLL LORAWAN_End_Node (in STM32Cub) >) Includes ✓ Application > △ App 	62 Build Configurations > Set Active > 1 Debua 63 Build Project Manage ✓ 2 Release 65 Build Working Set > Build by Working Set 67 Clean. ✓ 2 Release		 radio_board_if.h RBI_Init(void): int32_t RBI_Delnit(void): int32_t RBI_ConfigRFSwitch(RBLSN 	witch_Type(
Core Eartup Satrup_stm32wle5ccux.s Earget Aradii board if c	68 Build Automatically Set Active by Working Set 69 Build Targets Manage Working Sets 70 Build Targets Intervent = {0}; 71 Build Targets Total		 RBI_Gett XC ontig(void) : int RBI_GettWakeUpTime(void) RBI_IsTCXO(void) : int32_t RBI_IsDCDC(void) : int32_t 	s2_t : int32_t
 > be Drivers > be Utilities > be Utilities ↓ LoRaWAN_End_Node.ioc STM32WL55JCIX_FLASH.id 	73 Generate Report 74 Generate Report 75 Generate Code 76 Properties 77 Properties 78 gpio_init_structure.roue 79 gpio_init_structure.roue 79 gpio_init_structure.roue 79 gpio_init_structure.roue			
		~	<	>
		Duild Analys		
		Build Analyz	zer 🐹 🚊 Static Stack Analyzer	
	CDT Build Console [LoRaWAN_End_Node]			🛃 Ş
	^	Memory Regio	005 Memory Details	
		Region	Start address End address	c Si
		Kegion	Sant auuress Enu adures	, 3I.
< >	< >			>
C /LoRaWAN_End_Node/Application/Targ	et/radio_board_if.c			

Figure 44: Configure Build to Release

10. After setting the build configuration, you are now ready to build the project. You should see a successful compilation and generation of a .bin file.

BAK[®] Documentation Center

workspace_1.7.0 - LoRaWAN_End_Node	e/Applicat	ion/Targ	jet/radio_board_if.c	- STM32CubelD	E				- 0	×
File Edit Source Refactor Navigate	Search	Project	Run Window	Help						
📑 = 🔚 🐘 = 🗞 = 🐘 🔍 [0	@ : 😭	Op	pen Project			0			Q i 😭	EC MX
Project Explorer 22	R radie	Cle	ose Project			-	E	Dutline 😚 🛞 Bi	uild Targets	
	61	Bu Bu	uild All	Ctrl+B			^		a, 🔊 🔊 e	# 8
V DE LoRaWAN_End_Node (in STM32Cub	62	Bu	uild Configurations	>				radio_board_	f.h	
> 🔊 Includes	63	Bu	uild Project		dd the correspondent files in the IDE workspace */			 RBI_Init(void) 	: int32_t	
Application	65	Bu	uld Working Set	>	the preprocessor definitions or in platform.h */			 RBI_Delnit(vo RBI_ConfigRE 	id) : int32_t Switch/PPL Su	itch Tunol
> 🗁 App	67 -	Cle	ean					 RBI GetTxCor 	fiq(void) : int3	32 t
V 👝 Startup	68	Bu	uild Automatically		Init() but not supported by MX*/			RBI_GetWake	UpTime(void)	: int32_t
S startup_stm32wle5ccux.s	70	Bu	uild Targets	>	<pre>icture = {0};</pre>			RBI_IsTCXO(v RBI_IsTCXO(v)	oid) : int32_t	
Iarget Iarget Iarget difference	71	0	(C++ Index	、 、	тсн)			RBI_ISDCDC(\	roid) : int32_t	
> 🗁 Drivers	73	Ge	enerate Report		ack */					
> 👝 Middlewares	74	dh Ge	enerate Code							
> Utilities	76				pin */					
STM32WL55JCIX_FLASH.Id	78	Bbr	openies 20_1011_Structur	re.noue = u	PIO_MODE_OUTPUT_PP;					
	79 80	gpi	.o_init_structur	re.Pull = G	PIO_NOPULL; PIO_SPEED_EREO_VERY_HIGH·					
		<		crispeca - a		>	× <			>
	Prob	lems 🎜	Tasks 📄 Conso	le 🕸 🔲 Pror	perties	🗈 Build	d Analyzer	😤 🛋 Static Stac	k Analyzer	
						019		_		- 8
	CDT Build	CDT Build Console [LoRaWAN_End_Node]								
		^			Memo	ory Regions	ions Memory Details			
						Regio	n	Start address	End address	Si
						inegro				
					×	<				>
< >> (LoPaWAN End Node/Application (Trans	<	oard if -	-		>					
	ler/ladio_p	ioaru_if.c	-				:			

Figure 45: Build the Project



Figure 46: Successful Project Build

Running the LoRaWAN_End_Node Example of STM32WL SDK on RAK3172

Configuration to Connect the LoRaWAN Network Server

Once you have a working project and were able to build with no errors in the STM32CubeIDE, the next step is to configure the LoRaWAN parameters to be able to run **LoRaWAN_End_Node** example code with RAK3172.

 First, you need to register the device to the network server. You can follow the guide on how to register a device in TTN V3 or Chirpstack using the procedure discussed in the RAK3172 TTN V3 OTAA Quick Start Guide or in the RAK3172 Chirpstack OTAA Quick Start Guide respectively.

VOTE	
By default, the LoRaWAN_End_Node example will work on the EU868	region. This is set in the
lora_app.h that can be found in this location /STM32Cube_FW_WL_V1.	0.0/Projects/NUCLEO-
WL55JC/Applications/LoRaWAN/LoRaWAN_End_Node/LoRaWAN/App/ (for v	v1.0.0) and
/STM32Cube_FW_WL_V1.2.0/Projects/NUCLEO-	
WL55JC/Applications/LoRaWAN/LoRaWAN_End_Node/LoRaWAN/App/ (for v	v1.2.0).

2. To activate and connect your device via OTAA, you need to get the following parameters: **DEVEUI**, **APPEUI**, and **APPKEY**.

Once you successfully register your device to TTN V3, you should see those parameters, as shown in Figure 47.

RAK3172 ID: rak3172-init-te	est			
Last seen yesterday	2 🗸 n/a			
Overview Live data	Messaging Location Payload for	matters	Clair	ming General settings
General information				Live data
			E	
End device ID	rak3172-init-test			
Description	This end device has no description			
				Waitin
Created at	Mar 23, 2021 17:15:07			warch
Activation information				
AppEUI	11 11 11 11 99 99 99 99	<>		
DevEUI	22 22 22 22 88 88 88 88	<>		Location
Root key ID	n/a			
Notency is				
АррКеу	2B 7E 15 16 28 AE D2 A6 AB F7 15 88 09	0 CF 4F 30		
NwkKey	n/a			

Figure 47: OTAA device registration in TTN V3

3. With the device registered to TTN V3, you should edit the se-identity.h file to update the needed OTAA parameters. On the STM32CubeIDE, click **File** and select **Open File..**. You should navigate in this directory \STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-

WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN\App (for v1.0.0) or

\STM32Cube_FW_WL_V1.2.0\Projects\NUCLEO-

WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN\App (for v1.0.0) to find the se-identity.h, then click Open.

IDE	workspace 1.7.0 - STM32CubeIDE					- 0	×
File	Edit Source Refactor Navigate Searce	ch Project	Run Window Help				
	New Alt-	+Shift+N >	📴 र 🞯 र 🖗 र 🔕 र 💁 🖋 र 💷 💷 💷 👖 🖢 र 🖓 र एन्ट्र () र र न र 📑	0		Q 🔡	Ec MX
	Open File		.c 🛛		E Outline 🔀 💿 Bui	ild Targets	
	Open Projects from File System		R CODE END RBI_Init_1 */	^	🖻 J ^a	z 💘 🐋 🙍	# 8
	Recent Files	>	ined(USE_BSP_DRIVER) e generated by MX does not support BSP */		radio_board_if	.h	
	Close Editor	Ctrl+W	order to use BSP driver, add the correspondent files in the IDE workspace */		 RBI_Init(void) : RBI_Delnit(void) : 	int32_t 1): int32_t	
	Close All Editors Ctrl+	+Shift+W	BSP_RADIO_Init();		 RBI_ConfigRFS 	witch(RBI_Swi	.tch_Type[
	Save	Ctrl+S	etined(MX_NUCLEO_WL55JC1) uld be calling BSP RADIO Init() but not supported by MX*/		 RBI_GetTxConf 	ig(void) : int3	2_t
	Save As		TeitTureDef geis init stausture - (0).		 RBI_GetWakeU RBI_IsTCXO(vo 	plime(void) : id) : int32 t	int32_t
	Save All Ctrl	I+Shift+S	michypeben gpio_mic_schucture = {0};		 RBI_IsDCDC(volume 	oid) : int32_t	
	Revert		ined(RAK3172_RF_CHANNEL_SWITCH) nable the Radio Switch Clock */				
	Move		W_CTRL1_GPIO_CLK_ENABLE();				
	Rename	F2	onfigure the Radio Switch pin */				
<u>&</u>	Refresh	ر ا	init_structure.Pin = RF_SW_CTRL1_PIN;				
	Convert Line Delimiters to		init_structure.Pull = GPIO_NOPULL;				
	Print	Ctrl+P	CDTO Toit/DE SH CTD11 GDTO DOBT Proio init stausture).	× .			
2	Import						
	Export		lasks 🖢 Console 🐹 📄 Properties 👘 🖬 🖬 🖬 🖬 🖬 👘	Build Analyzei	r 23 🚊 Static Stack	Analyzer	
	Properties A	Alt+Enter					⊠ 8
	Switch Workspace	>		Memory Region	Managa Dataila		
	Restart			Region	Start address	End address	Si
_	Exit			Region	Start address	End budiess	51.
<	>			<			>

Figure 48: Open the File Needed to Modify with OTAA Parameters

ew folder			ie 🕶 🛄
Name	Date modified	Туре	Size
app_lorawan.c	28/10/2020 4:29 PM	C File	3 KB
🖻 app_lorawan.h	28/10/2020 4:29 PM	C/C++ Header	3 KB
CayenneLpp.c	28/10/2020 4:29 PM	C File	13 KB
🖻 CayenneLpp.h	28/10/2020 4:29 PM	C/C++ Header	3 KB
🗈 Commissioning.h	28/10/2020 4:29 PM	C/C++ Header	3 KB
lora_app.c	28/10/2020 4:29 PM	C File	21 KB
🖻 lora_app.h	28/10/2020 4:29 PM	C/C++ Header	4 KB
🖻 lora_app_version.h	28/10/2020 4:29 PM	C/C++ Header	3 KB
lora_info.c	28/10/2020 4:29 PM	C File	5 KB
🗟 lora info b	28/10/2020 4:29 PM	C/C++ Header	3 KB
🖻 se-identity.h	28/10/2020 4:29 PM	C/C++ Header	33 KB
File name: se-identity.h		×.*	

Figure 49: Open the se-identity.h File

4. The se-identity.h header file must be updated with the common DEVEUI, APPEUI, and APPKEY parameters in the device registration on the network server. In this example, you can see that the LORAWAN_DEVEUI_EUI, LORAWAN_JOIN_EUI, and LORAWAN_APP_KEY are updated, as shown in **Figure 50**. These values are based on the TTN V3 registration in **Figure 47**.

The LORAWAN_JOIN_EUI is the same as the App_EUI in this guide which is the term that adheres to the LORAWAN Specification V1.1.

NOTE

To ensure that your device work on both LoRaWAN versions (LoRaWAN Specifications V1.0.x and V1.1), make sure that the application rook key LORAWAN_APP_KEY and the network root key LORAWAN_NWK_KEY of the se-identity.h file are exactly the same. Else, you might encounter MIC-related errors while joining the network.



The macro **STATIC_DEVICE_EUI** is also updated to **1** instead of **0** since a generated DEVEUI in TTN V3 is used in this guide instead of the embedded DEVEUI of the device.

workspace_1.7.0 - D:\Downloads\en.st	tm32cubewl_v1.0.0\STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\	LoRaWAN_End_Node\LoRaWAN\App\se-identity.h - STM32CubelDE - 🛛 🗙
<u>File Edit Source Refactor Navigate</u>	Se <u>a</u> rch <u>P</u> roject <u>R</u> un <u>W</u> indow <u>H</u> elp	
📑 🗕 🔚 🐘 😵 🖛 🗞 👻	💩 🖆 🕶 📽 🕶 🖻 🕶 🚱 🕶 🚺 🕈 🔷 🕐 🖕 🌽 🖗 🔲 👖 🕍 🖛 🌾	│ + ♥ - ♥ - ♥ → → + 🛃 🚺 🔍 🤤 🔤 🔤
Project Explorer 💥 📃 🗖	c radio_board_if.c	
Image: Second Secon	<pre> radio_board_if.c term 'se-identity.h ☆ ret in the interval of the</pre>	<pre>t)9x91000000A</pre>
	122 * Network root key	
	<	>
	Problems 🔊 Tasks 📮 Console 🛿 🔲 Properties No consoles to display at this time.	■ ■ Build Analyzer ■ Static Stack Analyzer ■ ■ ■ ■ ■ ■ ■ Memory Regions Memory Details ■ ■ ■ Region Start address End address ≤
< >		٢ >
	Writable Smart	t Insert 92 : 1 : 3979

Figure 50: Modified se-identity.h File

📝 NOTE

These parameters are usually set and code generated in the **.ioc** file via STM32CubeMX. But that method is not possible since a direct modification of the radio-related source files is done in this guide. Any further code generation via STM32CubeMX after the modifications in the previous steps in this guide will override all the changes that are required to run the LoRaWAN_End_Node project example RAK3172 module.

Generation of BIN file

With all the needed files modified and edited, you can now generate your .bin FW file and upload it to your RAK3172 module.

 The first step is to clean first the project to remove any outdated binary files in the project folder then followed by building it. Sometimes **Build Project** is not clickable so you can use **Build All** as an alternative. You only have one project as of now so that should work fine as well.

workspace_1.7.0 - D:\Downloads\en.st	m32cubewl_v1.0.0\STM32Cube_FW_WL_V1.0.0\Proje	cts\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\LoRaWAN\App\se-identity.h - STM32CubelDE —		×
<u>File Edit Source Refactor Navigate</u>	Se <u>a</u> rch <u>Project Run W</u> indow <u>H</u> elp			
	🕼 👩 Open Project	▼ 🖄 🖋 ▼ 🗐 🗊 🗓 ▼ 🖗 ▼ 🖓 ▼ 🏷 🖓 🗢 ▼ 🖒 ▼ 📑 🚺 🚺 🔍 🔍	🗈 ا ا	EC MX
Project Explorer 🛛 🗖 🗖	Close Project		-	
	91 Build All Ctrl+B			A
✓ IDE LoRaWAN End Node (in STM32Cuł	92 # Build Configurations >	1		
> 🔊 Includes	93 949 / Build Project			
> 👝 Application	95 Build Working Set			
> 🔁 Drivers	96 97 # Clean	{ 0x22 0x22 0x22 0x22 0x28 0x88 0x88 0x88		
> 🔁 Middlewares	98 Build Automatically			
V Celedse	990 /			
LoRaWAN End Node.ioc	100 Build Targets >	ndian)		
STM32WL55JCIX_FLASH.Id	102 # MX Generate Report	{ 0x11, 0x11, 0x11, 0x11, 0x99, 0x99, 0x99, 0x99 }		
	103 1040 / db Generate Code			
	105	N_DEVICE_ADDRESS		
	106 Properties	tically set with a value provided by a pseudo		
	108 */	Aren a varue provided by che Meo practoria		
	109 #define STATIC_DEVICE_ADDRESS	0		
	110 1110 /*!			
	112 * Device address on the network	(big endian)		
	113 */	(uin+32 +)ava1aaaaa		
	115	(dintbi_) (overedeter		
	116⊖ /*!			
	117 * Application Poot key 118 */			
	119 #define LORAWAN_APP_KEY	2B,7E,15,16,28,AE,D2,A6,AB,F7,15,88,09,CF,4F,3C		
	120 <		>	
	💌 Problems 👘 Tarkr 🕞 Console 😚 🗔 Pro	natiar 🖓 🗖 🖻 Build Analyzar 🕅 📑 Static Stack Analyz	-	
			Lei	
	CDT Build Console [LoBaWAN End Node]			1 8
	epi balla console [constraint_ind_induc]			
		Memory Regions Memory Details		
		Region Start address End a	ddress	÷ .
		* <		>
< >>	<	· · · · · · · · · · · · · · · · · · ·		

Figure 51: Clean and Build the STM32CubeIDE Project

After a successful build, you should see in console Finished building: LoRaWAN_End_Node.bin. You should be able to see the generated LoRaWAN_End_Node.bin firmware file in this folder location
 \STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO

(31M32000C_1W_WL_V1.0.0(110)00003(N000L00-

 $wL55JC\Applications\LoRawAN\LoRawAN_End_Node\STM32CubeIDE\Release \ (for v1.0.0) \ or a statement of the s$

\STM32Cube_FW_WL_V1.2.0\Projects\NUCLEO-

WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\STM32CubeIDE\Release (for v1.2.0). That bin file is the firmware binary that you need to upload to your RAK3172 module.

workspace_1.7.0 - D:\Downloads\en.st	tm32cubewl_v1.0.0\STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Application	s\LoRaWAN\LoRaWAN_End_Node\LoRaWAN\App\se-identity.h - STM32Cubell	DE —		×	
<u>File Edit Source Refactor Navigate</u>	Se <u>a</u> rch <u>P</u> roject <u>R</u> un <u>W</u> indow <u>H</u> elp					
📑 🕶 🔚 🕞 😸 🕶 🗞 💌	🗄 🖆 = 😂 = 🖸 = 🧭 = 🎋 = 🖸 = 隆 = 🍠 =	□ 월 ▼ 禄 ▼ ♥ ♥ ♥ ♥ ♥ ♥	Q	8		
Project Explorer 🔀 📃 🗖	🖻 radio_board_if.c 🛛 🚡 se-identity.h 🔀					
E & ♥ 8 ↓ E LoRaWAN_End_Node (in STM32Cut ↓ ♥ Binaries ↓ Binaries ↓ Binaries ↓ Content ↓ Direction ↓ Direction ↓ Direction	91 */ 92 #define STATIC_DEVICE_EUI 93 94⊕ /*! 95 * end-device IEEE EUI (big endian) 96 */ 97 #define LORAWAN_DEVICE_EUI	1 { 0x22, 0x22, 0x22, 0x22, 0x88, 0x88, 0x88, 0x88 }			 ▲ ■ ■	
> > Middlewares > > Peterse > Comparison > Peterse + LoawAN_End_Node.ioc TM32WL55/CIX_FLASH.id	900 /*! 100 * App/Join server IEEE EUI (big endian) 101 */ 102 #define LORAWAN_JOIN_EUI 103	{ 0x11, 0x11, 0x11, 0x11, 0x99, 0x99, 0x99, 0x99 }				
	Image: Image					
	<pre>109 #define STATIC_DEVICE_ADDRESS 110 111⊕ /*! 112 * Device address on the network (big endian) 113 */ 114 #define LORAWAN_DEVICE_ADDRESS 115 116⊕ /*!</pre>	0 (uint32_t)0x0100000A				
	117 * Application root key 118 */ 119 #define LORAWAN_APP_KEY 120 <	28,7E,15,16,28,AE,D2,A6,A8,F7,15,88,09,CF,4F,3C		>	¥	
	😰 Problems 🧔 Tasks 🔤 Console 🖄 🛄 Properties	Build Analyzer 🛛 🚢 Static Stat	k Analyze	. –		
l	CDT Build Console [LoRaWAN_End_Node] LoRaWAN_End_Node.elf - Finished building: LoRaWAN_End_Node.bin ^/LoRaWAN_End_Node/Release - 15 Aug					
	Finished building: LoRaWAN_End_Node.list 00:08:54 Build Finished 0 errors, 1 warnings. (took 75.725ms) Memory Regions Memory Details Region Start address End address End address End address End address End address 0.000000 0.0080000 0.0080000 0.0080000 0.0080000 0.0080000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.008000000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.00800000 0.008000000 0.008000000 0.008000000 0.008000000 0.008000000 0.0080000000 0.008000000 0.008000000 0.008000000 0.008000000 0.008000000 0.008000000 0.008000000 0.008000000 0.00800000000					
< >	د	> <		>		

Figure 52: Successful Build with BIN File for RAK3172

Uploading the FW Generated Using STM32CubeProgrammer

The generated .bin FW file is ready to be uploaded to RAK3172.

To upload this binary file, you need to use STM32CubeProgrammer created by STMicroelectronics.

Download the latest version STM32CubeProgrammer 🗹 and the one compatible with your computer.

In this guide, you will use the internal UART bootloader of the STM32WL and connect the RAK3172 to a USB to Serial converter tool like the RAKDAP1. You need to connect five pins: power supply pins (3.3 V and GND), UART2 pins (TX and RX), and the Boot0 pin (connected to 3.3 V), as shown in **Figure 53**.



Figure 53: RAK3172 Connection to USB-UART Converter with Boot0 Pin at 3.3V Level

📝 NOTE

You can also use an ST-LINK to upload the .bin file to RAK3172.

1. Once the hardware is now ready, you can open the STM32CubeProgrammer. Then you need to open the LoRaWAN_End_Node.bin FW file from this folder location \STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO-WL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\STM32CubeIDE\Release (for v1.0.0) or \STM32Cube_FW_WL_V1.2.0\Projects\NUCLEO-

wL55JC\Applications\LoRaWAN\LoRaWAN_End_Node\STM32CubeIDE\Release (for v1.2.0).

Prg STIV	132CubeProgrammer		– 🗆 ×
STM32 Cube	Programmer (1997)	F 🕒	y 🛧 🖅
	Memory & File edition		Not connected
	Device memor Open file -	UART	▼ Connect
	Address Size Data width 32-bit • Read •	U	ART configuration
		Port	COM29 - Ø
OB		Baudrate	115200 -
		Parity	Even 👻
		Data bits	8 🗸
		Stop bits	1.0 👻
	No data to display	Flow control	Off 👻
	Log Verbosity level 1 2 3		
	00:41:03 : STM32CubeProgrammer API v2.1.0	(
		c	evice information
\odot		Device	-
		Туре	
		Device ID	
\bigcirc		CPU	
			-

Figure 54: STM32CubeProgrammer Application

V19 Open file X									
$\leftarrow \rightarrow$	A CoRaWAN_End_Node > S S S	TM32CubelDE → Release	5 V	🔎 Search Release					
Organize	 New folder 			:::					
<u>_</u> ^	Name	Date modified	Туре	Size					
	Application	13/08/2021 6:07 PM	File folder						
	Drivers	13/08/2021 6:07 PM	File folder						
	Middlewares	13/08/2021 6:07 PM	File folder						
r	Utilities	15/08/2021 12:08 AM	File folder						
	M LoRaWAN_End_Node.bin	15/08/2021 12:08 AM	Binary Log	62 KB					
	CORAWAN_ENG_NOGE.EII	15/08/2021 12:08 AM	ELF File	184 KB					
S									
S									
	File name: LoRaWAN_End_No	ode.bin	~	Firmware files (*.bin;*.binary;*.ł ∨					
				Open Cancel					

Figure 55: Locating the .bin Firmware File

2. If everything is successful, you should now see the LoRaWAN_End_Node.bin FW file in the STM32CubeProgrammer.

Pro STM	132CubeProgrammer								– 🗆 ×
STM32 Cube	Programmer						19	f 🖸 🕴	• 🛪 🏹
	Memory & File	edition							Not connected
	Device memory	LoRaWAN_End_	Node.bin × +					ST-LINK	Connect
	Address 0x0	-	Size 0x400	Data width	32-bit	•	Download 🔹	ST-LIN	IK configuration
	Address	٥	А	8	C	٨٩	I	Serial number	No ST-LIN 👻 💋
	0x00000000	20008000	08002009	08001881	08001883		â	Port	SWD 👻
	0x0000010	08001885	08001887	08001889	00000000			Frequency (kHz)	
	0x00000020	00000000	00000000	0000000	0800188B			Mode	
	0x0000030	0800188D	00000000	0800188F	08001891			Mode	Normal 👻
	0x00000040	08002059	08002059	08001895	08002059	YYY		Access port	0 -
	0x00000050	08002059	08002059	080018A1	080018A7	YYj§		Reset mode	Software reset 🔹 👻
	0x00000060	08002059	08002059	08002059	08002059	YYYY		Shared	Dirabled
	0x00000070	08002059	08002059	08002059	080018AD	Y Y Y			Disabled
	0x00000080	08002059	08002059	08002059	08002059	Y Y Y Y		External loader	-
	0x00000090	00000000	08002059	08002059	08002059	YYY		larget voltage	-
	0x000000A0	08002059	08002059	08002059	08002059	Y Y Y Y		Firmware version	-
	0x00000080	08002059	08002059	08002059	08002059	Y Y Y Y			Firmware upgrade
	0x000000000	08002059	08002059	08002059	08002059	YYYY	~		
	Log					Verbosity level	1 2 3		
	00:23:13 : STM32CubeProgrammer API v2.1.0 00:23:19 : Read File: D:\DownloadS\en.stm32cubeVl_v1.0.0\STM32Cube_FW_WL_V1.0.0\Projects\NUCLEO- WL552ClApplications\LokAWAN_Ln_dNode\STM32CubeIDE\Release\LokAWAN_End_Node.bin 00:23:19 : segments: 1 00:23:19 : segments: 0): address= 0x0, size= 0xF6B0							Devi	ce information
Θ								Device	-
\bigcirc								Туре	-
Ø								Device ID	-
\bigcirc								Flash size	-
$\langle \cdot \rangle$								CPU	-

Figure 56: Loaded .bin Firmware File to the STM32CubeProgrammer

3. By default, the STM32CubeProgrammer chooses ST-LINK as the uploading interface, so you need to change it to UART and select the right COM port. After setting up the UART connection, you can now connect and see that the device is detected.

Prg STM	132CubeProgrammer								-	
STM32 Cube	• 🍞 Programmer						(19)	f 🕒	¥ 🔀	57
	Memory & File	edition							Not 🔴	connected
	Device memory	.oRaWAN_End	Node.bin × +					UART	• (onnect
	Address 0x0	*	Size 0x400	Data width	32-bit	• Downlo	ad 👻	ST-LINK UART	fguration	
	Address	0	4	8	С	ASCII		USB	DM29	- S
	0x00000000	20008000	08002009	08001881	08001883		â	OTA	.5200	
	0x0000010	08001885	08001887	08001889	00000000			Parity	Even	
	0x0000020	00000000	00000000	00000000	0800188B		U		Even	_
	0x0000030	0800188D	00000000	0800188F	08001891			Data bits	8	×.
	0x0000040	08002059	08002059	08001895	08002059	ΥΥΥ		Stop bits	1.0	v
	0x0000050	08002059	08002059	080018A1	080018A7	YYį§		Flow control	Off	
	0x0000060	08002059	08002059	08002059	08002059	Y Y Y Y				
	0x0000070	08002059	08002059	08002059	080018AD	YYY				
	0x0000080	08002059	08002059	08002059	08002059	YYYY				
	0x0000090	00000000	08002059	08002059	08002059	YYY				
	0x000000A0	08002059	08002059	08002059	08002059	YYYY				
	0х00000в0	08002059	08002059	08002059	08002059	Y Y Y Y				
	0x000000C0	08002059	08002059	08002059	08002059	YYYY	~			
	Log					Verbosity level 💿 1 💮 :	2 🔘 3			
(3) (2) (2)	00:23:13 : STM3; 00:23:19 : Read WL55JC/Applicat 00:23:19 : Numb 00:23:19 : segme	2CubeProgramm File: D:\Dow ions\LoRaWAN\ er of segment ent[0]: addre	er API v2.1.0 mloads\en.stm32c. LORawAN_End_Node :s: 1 :ss= 0x0, size= 0:	ubew]_v1.0.0\STM32 \STM32CubeIDE\Rele xF6B0	Cube_FW_WL_V1 ase\LoRaWAN_E	L.O.O'Projects\NUCLEO- nd_Node.bin	*	D Device Type Device ID Flash size CPU	evice information	

Figure 57: Selecting UART as Programming Interface

⁰ STM	132CubeProgramme	r							-	
TM32 Cube	Programmer						19	f 🕒	⊻ 🔀	57
	Memory & F	ile edition							🛑 Not o	connected
	Device memor	y LoRaWAN_En	d_Node.bin × +					UART	• C	lonnect
	Address 0x	0 -	Size 0x400	Data width	32-bit	•	Download 🔹	UA	ART configuration	
	Address	0	4	8	С	ASCII		Port	COM29	- Ø
7	0x0000000	20008000	08002009	08001881	08001883			Baudrate	COM29	
В	0×00000010	08001885	08001887	08001889	00000000			Darity	COM14	_
	0x0000020	0000000	00000000	00000000	0800188B				COM30	
	0x0000030	08001880	00000000	0800188F	08001891			Data bits	COM16	-
	0x00000040	08002059	08002059	08001895	08002059	YYY		Stop bits	1.0	-
	0x00000050	08002059	08002059	080018A1	080018A7	YYj§		Flow control	05	
	0x0000060	08002059	08002059	08002059	08002059	YYYY		rion control	Un	
	0x00000070	08002059	08002059	08002059	080018AD	YYY				
	0x0000080	08002059	08002059	08002059	08002059	Y Y Y Y				
	0x0000090	0000000	08002059	08002059	08002059	YYY				
	0x000000A0	08002059	08002059	08002059	08002059	Y Y Y Y				
	0х00000в0	08002059	08002059	08002059	08002059	YYYY				
	0x000000C0	08002059	08002059	08002059	08002059	Y Y Y Y		~		
	Log					Verbosity level	1 2 3			
3	00:23:13 : S 00:23:19 : R WL55JC\Appli 00:23:19 : N 00:23:19 : S	TM32CubeProgram ead File: D:\Dc cations\LoRaWA umber of segmer egment[0]: addr	mer API v2.1.0 wwnloads\en.stm32ci ULoRaWAN_End_Node ts: 1 ess= 0x0, size= 0	ubewl_v1.0.0\STM32 \STM32CubeIDE\Rele xF6B0	Cube_FW_WL_V1 ase\LoRaWAN_E	.0.0\Projects\NUCLEO- nd_Node.bin	*	D Device	evice information	n
2								Device ID		
2								Electrice ID		
2	L							CDU		
\mathcal{I}								CPU		

Figure 58: Selecting the Right COM Port

4. You need to ensure that the Boot0 is connected to VDD (3.3 V) when the device is powered up, else, the STM32CubeProgrammer might not detect the device. The logs of a detected device are shown in **Figure 59**.

STM32 Cube	Programmer							<u>(19)</u>	f 🕒 🕚	• 🛧 🖅
=	Memory & File		Connected							
	Device memory	LoRaWAN_End_Ncc	le.bin× +						UART	Disconnect
	Address 0x0	▼ Size	0xF6B0	Data width	32-bit 🔻	Find Data	Ox	Download 🔻	UAR Port	
	Address	0	4	8	(ASCI	I	Baudrate	445000
OB	0x0000000	20008000	08002009	08001881	080018	83		Â		115200
	0x00000010	08001885	08001887	08001889	000000	00			Parity	Even 👻
CPU	0x00000020 0000000 0000		00000000	00000000	080018	8B			Data bits	8 -
	0x0000030	0800188D	00000000	0800188F	080018	91			Stop bits	
swv	0x00000040	08002059	08002059	08001895	080020	59 Y.	YY			
	0x0000050	08002059	08002059	080018A1	080018	A7 Y.	Y;§		Flow control	Off 🔻
	0.00000000	00000050	00000050	00000050	000000			×	RTS	0
	Log					Live Update	Verbosity level	1 2 3	DTR	0
	00:57:16 : DTR low	teaasbachid 0001 is see	occording opened.				-	^ 🖪	Dead Linnratest (
	00:57:17 : Port confi 00:57:17 : Activation	guration: parity = even, ba	udrate = 115200, da	ta-bit = 8,	stop-bit = 1.0, fl	ow-control = of	f		Read Unprotect (wco)
	00:57:17 : Chip ID: 0	x497							Tar Board	get information
	00:57:17 : BootLoad 00:57:17 : UPLOAD	fer protocol version: 3.3 ING OPTION BYTES DAT/	A						Device	STM32WLxx
	00:57:17 : Bank	: 0x00 : 0x1fff7800							Device ID	0x497
	00:57:17 : Size	: 104 Bytes							Revision ID Flash size	 256 KB - Default
æ	00:57:17 : UPLOAD 00:57:17 : Size	NG : 1024 Bytes							CPU Bootloader Versio	Cortex-M4
W	00:57:17 : Address	:0x8000000								
()	00:57:17 : Data read	successfully								
	00:57:17 : Time elap	sed during the read opera	tion is: 00:00:00.13	0				~		
(?)										
\sim								100% 🛇		

Figure 59: RAK3172 Detected by STM32CubeProgrammer

5. If the device is detected by the STM32CubeProgrammer, you can now upload the firmware by clicking **Download**.

STM32 Cube	Programmer								f 🖸 🔰	• 🛪 🖅			
	Memory &	File ed	dition							Connected			
	Device memory LoRaWAN_End_Ncde.bin× +									Disconnect:			
<u> </u>	Address	0x0	▼ Size	0xF6B0	Data width 32-	bit: 🔻 Find D	ata Ox	Download 🔻	UAR Port	T configuration			
	Addre	ess	0	4	8	С	ASCII		Baudrate				
OB	0x000000	00	20008000	08002009	08001881	08001883		^	Baadiato	115200			
	0x000000	10	08001885	08001887	08001889	00000000			Parity	Even 👻			
CPU	Ox0000020		0000000	00000000	0000000	0800188B			Data bits	8 🔻			
(cum)	0x0000030		0800188D	00000000	0800188F	08001891			Stop bits	10			
SWV	0x0000004	40	08002059	08002059	08001895	08002059	YYY						
	0x000000	50	08002059	08002059	080018A1	080018A7	YYj§		Flow control	Off 🔻			
×			00000050	00000050	00000050	00000050			RTS	0 👻			
	Log		124 Puton			Live U	odate Verbosity level	1 2 3	DTR	0 -			
	00:45:01: A	ddress	: 0x8000000						Read Linnrotect (1	Read Linprotect (MCLI)			
	00:45:01 : Re 00:45:01 : Da	ad progres ta read su	ss: ccessfully										
	00:45:01 : Tin	ne elapseo	d during the read operations and operation of the second operation operat	tion is: 00:00:00.11) bin				Board				
	00:45:14 : Nu	mber of s	egments: 1	RavvAin_Enu_inoue	.om				Device Type	STM32WLxx MCU			
	00:45:14 : seg 00:47:40 : Me	ment[0]: mory Pro	address= 0x0, size= 0: gramming	xF6B0					Device ID Revision ID	0x497			
UETA	00:47:40 : Op	iening and	d parsing file: LoRaWAN	LEnd_Node.bin					Flash size	256 KB - Default			
	00:47:40: S	ze :6	3152 Bytes						Bootloader Versio	n			
	00:47:40 : A 00:47:40 : Era	ddress Ising men											
(\mathbf{a})	00:47:40 : Era	sing inter wnload in	ral memory sectors [0 Progress:	30]									
) >					
(?)								71% 🗙					

Figure 60: Firmware Uploading in Progress

STM32 Cube	Programme	er											f 🕨	y	\star	77
=	Memory	& File e	dition											•	Connec	cted
	Device m	emory L	.oRaWAN_	End_Ncc	le.bin × +								UART	•	Disco	nnect
	Address	0x0	•	Size	0xF6B0	Data width	32-	bit: 🔻	Find Data	0x		Download 🔻	Port	UART confi	guration usbs	- 0
	Address			0	4	8	8		;		ASCII		Baudrate	115	200	
OB	0x00000	000	200080	000	08002009	08001881		08001883				î	Des Stat		2000	
	0x00000	010	080018	885	08001887	08001889	000000						Parity	Eve	in	×
СРО	0x00000020 00000000		00000000	00000000	0000000		3B				Data bits	8		-		
swv	0x00000030 0800188D 0000		00000000	• • •		Prg	Message				Stop bits	1.0		-		
	0x00000040 08002059			08002059	- File	down	load compl	oto				Flow cont	ral ov			
	0x00000050 08002059 08002059										~					
<u> </u>	Log											2 3	RTS	0		· ·
	00:57:17:0PLOADING									UK		DTR	0		-	
	00:57:17 : 00:57:17 :	Size : 1 Address	024 Bytes : 0x800000	0								- 3	Read Unp			
	00:57:17 : F	Read progre	ISS:											Targetinfo	ormation	
	00:57:17 : 1	lime elapse	d during the	read opera	tion is: 00:00:00.13	5							Board		ST	 M32WI ¥¥
	00:58:58 : 00:58:58 :	Memory Pro Opening an	ogramming Id parsing file	: : LoRaWA	N End Node.bin								Type			MCU
	00:58:58 :	File : L	oRaWAN_Er	nd_Node.b	in								Revision II	D		0x497
	00:58:58 :	Address	: 0x080000	000									CPU		256 Ki	3 - Default Cortex-M4
	00:58:58 : 00:58:58 :	mory corresp rnal memory	sectors [0	30]								Bootloade	rVersion			
	00:58:59 : 00:59:07 :	Download in File downloa	n Progress: ad complete													
	00:59:07 :	Time elapse	ed during dov	vnload ope	ration: 00:00:08.80)7										
\bigcirc												×				
\odot												100% 😣				

Figure 61: Firmware Successfully Uploaded

6. After the successful download, restart the device and remove the connection of the Boot0 pin to VDD (3.3 V), leaving you only with four-pin connections (power supply lines and UART2) as shown in **Figure 62**.



Figure 62: RAK3172 Connection to UART

7. By using Serial Terminal software, check the serial output logs of the RAK3172 with the newly uploaded FW with baud rate setting 115200. You should be able to see the serial logs, as shown in **Figure 63**.

APP_VERSION:		V:	L.Ø.	0												
MW_LORAWAN_VER	SION	: V2	2.2.	1												
MW_RADIO_VERSION: V0.6.1																
###### OTAA ######																
###### AppKf±APP_VERSION: V1.0.0																
MW_LORAWAN_VERSION: V2.2.1																
MW_RADIO_VERSION: V0.6.1																
###### OTAA ##	####															
###### AppKey:	11	11	11	11	11	11	11	11	66	66	66	66	66	66	66	66
###### NwkKev:	2B	7E	15	16	28	AE	D2	A6	AB	F7	15	88	09	CF	4F	3C
###### ABP ##	####															
###### AppSKey	: 2B	7E	15	16	28	AE	D2	A6	AB	F7	15	88	09	CF	4F	3C
###### NwkSKey	: 2B	7E	15	16	28	AE	D2	A6	AB	F7	15	88	09	CF	4F	3C
###### DevEui:	22	-22-	-22-	-22-	-88-	-88-	-88-	-88								
###### AppEui:	11	-11	-11-	11.	-99-	.99.	-99-	-99								
0s012:TX on fr	eg 8	6830	0000	90 H	tz (at [DR ()								
1s496:MAC txDo	ne															
6s518:RX_1 on	frea	868	3300	0000) На	z at	t DF	x 0								
6s645:PRE 0K																
7s191:HDR OK																
8s337:MAC rxDo	ne															
###### = JOINE	D = (ОТА/	۹ ==													
10s014:temp= 2	9															
10s014:VDDA= 2	54															
10s015:TX on f	rea	8675	5000	000	Hz	at	DR	0								
10s016:SEND RE	OUES	г						-								
11s664 · MAC +xD	one															
16s685:RX 1 on	fre	a 86	5750	0000)0 F	tz (nt [DR Ø	,							
16s812 PRF OK		9 01														
17s358 HDR OK																
17s849:MAC ryD	one															
2. 55 15 INAC 1 XD	0															
###### =======		MCPS	5-00	onfi	irm											
20s014:temp= 3	0															
eenip 0	-															

Figure 63: RAK3172 UART2 Logs

8. With the device registered to TTN, you should now see a successful join and LoRaWAN device uplink.

	My PAK Work Applications	Applications > My RAK Wor	k Applications > End devices > RAK3172				
-	my tak work applications						
55	Overview	ID: rak3172-init-te	est				
X	End devices	• Last seen 1 second ago	↑2 ↓ n/a				Created 144 days ago
ıL	Live data	Overview Live data	Messaging Location Payload formatters	Clain	ning General se	ettings	
<>	Payload formatters	General information			• Live data		See all activity →
大	Integrations ~	End device ID	rak3172-init-test		↑ 01:01:21	Forward uplink data messa	ge Payload: { } 00 27 10 1D 0:
	Collaborators	Description	This end device has no description		↑ 01:01:12	Forward uplink data messa	ge Payload: { } 00 27 10 1D 0:
От	API keys	Created at	Mar 23, 2021 17:15:07		<i>≡</i> 01:00:54	The events list has been clear	
\$	General settings	Activation information					
		AppEUI	11 11 11 11 99 99 99 99 49				
		DevEUI	22 22 22 22 88 88 88 88 88 88		Location		Change location settings \rightarrow
		Root key ID	n/a				
_ с н	ide sidebar	АррКеу	·····	0			1.0.0

Figure 64: RAK3172 in TTN V3 Join and Uplink

Last Updated: 3/15/2023, 2:27:00 PM