RAK1906 WisBlock Environmental Sensor Datasheet

Overview

Description

The RAK1906 WisBlock Environmental Sensor module, part of the RAK WisBlock Sensor series, is a 4-in-1 digital sensor board that comprises gas, humidity pressure, and temperature sensor based on the Bosch® BME680 module. The RAK1906 is ideal for applications such as indoor air quality, home automation, and building IoT solutions.

Features

- Temperature range: -40 °C to 85 °C
- Humidity range: 0 to 100%
- Pressure range: 300 hPa to 1100 hPa
- Gas sensor response time: < 1 sec
- Gas sensor output: direct output of IAQ (Indoor Air Quality) index
- Module size: 10 x 10 mm

Specifications

Overview

Mounting

The RAK1906 module can be mounted on the slots: A, B, C, or D of a WisBlock Base board. Figure 2 shows the mounting mechanism of the RAK1906 on a WisBlock Base board, such as the RAK5005-O.



Figure 1: RAK1906 WisBlock Environmental Sensor Mounting

Hardware Chipset

Vendor

Part number

BOSCH

BME680

Pin Definition

The RAK1906 WisBlock Environmental Sensor module comprises a standard WisIO connector. The WisIO connector allows the RAK1906 module to be mounted on a WisBlock Base board, such as RAK5005-O. The pin order of the connector and the definition of the pinout are shown in Figure 3.

NOTE:

Only the I2C related pin, VDD, and GND are connected to this module.



Figure 2: RAK1906 WisBlock Environmental Sensor Pinout

Sensors

Temperature Sensor

Parameter	Symbol	Condition	Min	Тур	Max	Unit
Operating Temperature Range	T _A	Operational	-40	25	85	°C
Supply Current	I _{DD,T}	1 Hz Forced Mode, Temperature Measurement Only		1.0		μΑ
A _{T,25} Absolute Accuracy	A _{T,25}	25 °C		±0.5		°C
Temperature	A _{T,full}	0-65 °C		±1.0		°C
Output Resolution	R _T	API Output Resolution		0.01		°C

Humidity Sensor

BAK[°] Documentation Center

Parameter	Symbol	Condition	Min	Тур	Мах	Unit
			-40	25	85	°C
Operating Range			0		100	% r.H.
Full Accuracy Range			0		65	°C
			10		90	% r.H.
Supply Current	I _{DD,H}	1 Hz Forced Mode, Temperature and Humidity Measurement		2.1	2.8	μΑ
Absolute Accuracy	A _h	20-80% r.H., 25 °C, including hysteresis		±3		% r.H.

Pressure Sensor

Parameter	Symbol	Condition	Min	Тур	Max	Unit
Operating Temperature	T.	Operational	-40	25	85	°C
Range	۲A	Full Accuracy	0		65	C
Operating Pressure range	Ρ	Full Accuracy	300		1100	hPa
Supply Current	I _{DD,LP}	1 Hz Forced Mode, Pressure and Temperature, Lowest Power		3.1	4.2	μΑ
Temperature Coefficient	TCOp	25-40 °C 900 bPa		±1.3		Pa,K
Of Offset	reop	25-40°C, 900 HF a		±10.9		cm/K
Absolute Accuracy Pressure	A _{p,full}	300-1100 hPa 0-65 °C		±0.6		hPa

AQI Sensor

BAK[°] Documentation Center

Parameter	Symbol	Condition	Min	Тур	Max	Unit
Accuracy Status	AX _{IAQ}	Android Compatible	0		3	
IAQ Resolution	IAQ _{rs}			1		
IAQ Range	IAQ _{rg}		0		500	
Sensor-to-sensor Deviation	IAQ _{S2S}	All Operation Modes		±15% ±15		
Durability To	IAQ _{S2S}	Sensor-to-sensor Deviation		±15% ±15		
Siloxanes	IAQ _{drift}	Drift at Low & High Concentrations		±1% ±4		

Electrical Characteristics Recommended Operating Conditions

Symbol	Description	Min.	Nom.	Max.	Unit
V _{DD}	Power supply for the module	1.71	1.8	3.6	V
I _{sleep}	Sleep current	-	0.15	-	uA
I _{DD}	Humidity Measure current	-	340	-	uA
I _{DD}	Pressure Measure current	-	714	-	uA
I _{DD}	Temperature Measure current	-	350	_	uA

Mechanical Characteristics

Board Dimensions

Figure 3 shows the dimensions and the mechanic drawing of the RAK1906 module.



Figure 3: RAK1906 WisBlock Environmental Sensor Mechanic Drawing

WisConnector PCB Layout



Figure 4: WisConnector PCB footprint and recommendations

Schematic Diagram

Figure 5 shows the schematic of the RAK1906 module.



Figure 5: RAK1906 WisBlock Environmental Sensor schematics

RAK1906 Quick Start Guide

Prerequisite

What Do You Need?

Before going through each and every step on using RAK1906 WisBlock module, make sure to prepare the necessary items listed below:

Hardware

- RAK1906
- Your choice of WisBlock Base I
- Your choice of WisBlock Core
- USB Cable
- Li-Ion/LiPo battery (optional)

Software

- Download and install Arduino $\mathsf{IDE}\,\square$.
- Install RAKwireless Arduino BSP and add the RAKwireless Core boards to your **Arduino Boards Manager**. Follow the steps in the Github repo □ .

Product Configuration

Hardware Setup

WisBlock can integrate this module which makes it easy to build up an environmental temperature and humidity data acquisition system.

For more information about RAK1906, refer to the Datasheet.

The RAK1906 module gives information about:

- Air pressure
- Air quality
- Ambient Air Temperature
- Environment Humidity

RAK1906 module can be connected to any slot of WisBlock Base to communicate with the WisBlock Core. It will work on **SLOT A to D**. Also, always secure the connection of the WisBlock module by using the compatible screws.



Figure 1: RAK1906 connection to WisBlock Base

Assembling and Disassembling of WisBlock Modules Assembling

As shown in **Figure 2**, the location for Slot A, B, C, and D are properly marked by silkscreen. Slots C and D are located on bottom of WisBlock Base. Follow carefully the procedure defined in RAK5005-O module assembly/disassembly instructions 12 to attach a WisBlock module. Once attached, carefully fix the module with a M1.2 x 3 mm screw.



Figure 2: RAK1906 connection to WisBlock Base

Disassembling

The procedure in disassembling any type of WisBlock modules is the same.

1. First, remove the screws.



Figure 3: Removing screws from the WisBlock module

2. Once the screws are removed, check the silkscreen of the module to find the correct location where force can be applied.



Figure 4: Detaching silkscreen on the WisBlock module

3. Apply force to the module at the position of the connector, as shown in **Figure 5**, to detach the module from the baseboard.



Figure 5: Applying even forces on the proper location of a WisBlock module

📝 NOTE

If you will connect other modules to the remaining WisBlock Base slots, check on the WisBlock Pin Mapper I[™] tool for possible conflicts. RAK1906 uses I2C communication lines, and it can cause possible conflict especially on some IO modules.

After all this setup, you can now connect the battery (optional) and USB cable to start programming the RAK1906.

WARNING

- Battery can cause harm if not handled properly.
- Only 3.7-4.2 V Rechargeable LiPo batteries are supported. It is highly recommended not to use other types of batteries with the system unless you know what you are doing.
- If a non-rechargeable battery is used, it has to be unplugged first before connecting the USB cable to the USB port of the board to configure the device. Not doing so might damage the battery or cause fire.
- Make sure the battery wires match the polarity on the RAK WisBlock Base Board. Not all batteries have the same wiring.
- Only 5 V solar panels are supported. Do not use 12 V solar panels. It will destroy the charging unit and eventually other electronic parts.

Software Configuration and Example

The RAK1906 is a 4-in-1 Environmental Sensor board that contains the Bosch BME680 chip. It is specially designed for wearables where size and low power consumption are critical requirements. The BME680 contains a small Metal-Oxide (MOX) sensor and an I2C interface used to communicate with WisBlock Core.

WARNING

The BME680 requires a burn-in period to collect accurate readings. The burn-in procedure can be achieved by reading all sensors once every 5 seconds for at least 20 minutes the first time that you use it. For each subsequent use, the readings should stabilize 2-3 minutes after power-up.

Initial Test of the RAK1906 WisBlock Module

If you already installed the RAKwireless Arduino BSP \square , the WisBlock Core and example code should now be available on the Arduino IDE.

1. First, you need to select the WisBlock Core you have, as shown in Figure 6 to Figure 8.

RAK4631 WisBlock Core

RAK4631 Wisblock Core



Figure 6: Selecting RAK4631 as WisBlock Core

RAK11200 WisBlock Core

🥺 sketch_apr26a A	rduino 1.8.12			
File Edit Sketch To	ols Help			
	Auto Format	Ctrl+T		
	Archive Sketch			
sketch_apr26a	Fix Encoding & Reload			
<pre>void setup()</pre>	Manage Libraries	Ctrl+Shift+I		
// put your	Serial Monitor	Ctrl+Shift+M		
3	Serial Plotter	Ctrl+Shift+L		
	WiFi101 / WiFiNINA Firmware Updater			
// put your	ESP8266 Sketch Data Upload			
}	Board: "WisCore RAK11200 Board"		Boards Manager	
	Upload Speed: "921600"			
	Flash Frequency: "80MHz"		XinaBox CW01	
	Flash Mode: "QIO"		ESPrezzo Lite 1.0	
	Partition Scheme: "Default 4MB with spiffs (1,2MB APP/1,5MB SPIFFS)		ESPresso Lite 2.0	
	Core Debug Level: "None"		Discrete 1.0	
	Dort		Phoenix 1.0	
	Get Reard Infe		Phoenix 2.0	
	der board mito		NodeMCU 0.9 (ESP-12 Module)	
	Programmer: "AVRISP mkll"		NodeMCU 1.0 (ESP-12E Module)	
	Burn Bootloader		Olimex MOD-WIFI-ESP8266(-DEV)	
			SparkFun ESP8266 Thing	
			SparkFun ESP8266 Thing Dev	
			SparkFun Blynk Board	
			SweetPea ESP-210	
			LOLIN(WEMOS) D1 R2 & mini	
			LOLIN(WEMOS) D1 mini Pro	
			LOLIN(WEMOS) D1 mini Lite	
			WeMos D1 R1	
			ESPine (ESP. 12 Module)	
			TheiGen Class's COline	
			Thaleasyclec's Espino	
			winno	
			Arduno	
			4D Systems gen4 IoD Range	
			Digistump Oak	
			WiFiduino	
			Amperka WiFi Slot	
			Seeed Wio Link	
			ESPectro Core	
			Schirmilabs Eduino WiFi	
			ITEAD Sonoff	
			DOIT ESP-Mx DevKit (ESP8285)	
	· · · · · · · · · · · · · · · · · · ·		RAKwireless ESP32 Modules	
			WisCore RAK11200 Board	
			RAKwireless nRF Modules	
1			WisBlock Core RAK4631 Board	
			MicPlack Care PAK4601 Peard	

Figure 7: Selecting RAK11200 as WisBlock Core

RAK11310 WisBlock Core

 Blink Arduino 1.8.15 File Edit Sketch Tools Help 				- 6	۶ ×	
Auto Format Ctrl+T Archive Sketch Blink Fix Encoding & Reload					@ ▼	
1 /* Manage Libraries Ctrl+Shift+1 2 Blink Serial Monitor Ctrl+Shift+M 3 Turns an Serial Plotter Ctrl+Shift+L 4 Turns an Serial Plotter Ctrl+Shift+L 6 Most Ard WiFi101 / WiFiNINA Firmware Updater WiFi101 / WiFiNINA Firmware Updater 7 it is at Board: "WisBlock RAK11300" Serial Plotter 9 If you and the corr Port: "COM8 (WisBlock RAK11300)" Serial Plotter 10 model, c Get Board Info Serial Plotter	, repeatedly. the UNO, MEGA and ZERO Boards Manager Arduino AVR Boards Bouffalo Arduino Core					^
11 https:// 12 https:// 13 modified 14 by Scott 15 modified 2 Sep 2016 16 by Aturo Guadaupi 17 modified 8 Sep 2016 18 by Colby Newman 19 10 20 This example code is in the public domain. 21 https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blight 22 */	ESP32 Arduino > ESP32 Arduino (in sketchbook) > RAKwireless ESP32 Modules > RAKwireless RRF Modules > Rakwireless Raspberry Modules > STM32 boards groups (Board to be selected from Tools submenu 'Board part number') >	WisBlock RAK11300				
<pre>24 25 // the setup function runs once when you press reset or 26 void setup() { 27 // initialize digital pin LED_BUILTIN as an output. 28 pinMode(LED_BUILTIN, OUTPUT); 29 } 30 31 // the loop function runs over and over again forever</pre>	power the board					~
30			WisBI	ook RAK11300	I on COM8	

Figure 8: Selecting RAK11310 as WisBlock Core

2. The RAK1906_Environment_BME680 example ☐ in Github will work on all WisBlock Core. You can open the example codes depending on your WisBlock Core, as shown in Figure 9 to Figure 11.

💿 I File	RAK1906_Enviro	onment_BME680	Arduino 1.8.13				-	٥	×
riie	New	Ctrl+N							
	Open	Ctrl+0							20-
	Open Recent		, , , ,						•
	Sketchbook	;	>						^
	Examples	3	▲						
	Close	Ctrl+W	Examples for any board						
	Save	Ctrl+S	Bridge	>	[PAK12002 FID MI VO0522	1		
	Save As	Ctrl+Shift+S	Ethernet	>		RAK12003_FIR_MLX90032			
			Firmata	>		RAK12000_GPS_ZOE-MI8Q			
	Page Setup	Ctrl+Shift+P	LiquidCrystal	>		RAKISOOO_EEPROM_A124C02			
	Print	Ctrl+P	SD	>		RAK15001_Flash_GD25Q16C			
	Preferences	Ctrl+Comma	Stepper	>		RAK18000_Stereo			
			Temboo	> arymanager/Al	l#Adafruit_BME680	RAK18001_Buzzer			
	Quit	Ctrl+Q	RETIRED	>		RAK19002_Boost_IPS61046			
//	Might need	adjustments				RAK1901_Temperature_Humidity_SHTC3			
≢de	efine SEALEV	ELPRESSURE_H	Examples for WisBlock Core RAK4631 Board			RAK1902_Pressure_LPS22HB			
		1.0	Adafruit Bluefruit nRF52 Libraries	>		RAK1903_Optical_OPT3001			
1	La Dme680_1h	110()	Adafruit Internal File System on Bluefruit nRF52	>		RAK1904_Accelerate_LIS3DH			
1.	/ire.begin()	;	nRF52840 PDM - Adafruit Fork	>	1	RAK1906_Environment_BME680			
			RAK WisBlock examples	> RAK4601 >		RAK1910_GPS_UBLOX7			
1	if (!bme.beg	<pre>in(0x76)) {</pre>	Rotary Encoder	RAK4631	Communications	RAK1920_Grove_Color_TCS3472			
	return:	incin ("Courd	Servo	> 31) 1	10 >	RAK1920_Grove_PIR_AS312			
	}		SoftwareSerial	>	Power >	RAK1920_MikroBUS_Temperature_TMP102			
			SPI	>	Sensors >	RAK1920_QWIIC_AirQuality_SGP30			
	<pre>// Set up ov me setTempe</pre>	versampling a	Wire	>	Solutions >				
1	ome.setHumid	lityOversampl	Examples from Custom Libraries						
ł	ome.setPress	sureOversampl	Adafruit Circuit Playground	>					
1	ome.setIIRFi	lterSize(BME		>					~
			SparkEup ATECCX08a Arduino Librany	>					
								_	
1						WisBlock Core	RAK4631 Board, 0.3.2 SoftDevice s140 6.1.1, Level 0 (R	elease) on (COM6

Figure 9: Opening RAK1906 example code for RAK4631 WisBlock Core

New	Ctrl+N				1	
Open	Ctrl+O					~
Open Recent	>					-
Sketchbook	>					-
Examples	2	▲				
Close	Ctrl+W	HTTPUpdate	>			
Save	Ctrl+S	HTTPUpdateServer	>			
Save Ar	Ctrl+Shift+S	NetBIOS	>			
Save As	Cur+Shirt+S	Preferences	>			
Page Setup	Ctrl+Shift+P	RAK WisBlock examples	Communications >			
Print	Ctrl+P	SD(esp32)	IO >			
Preferences	Ctrl+Comma	SD_MMC	Power >			
- Tereneters	cur comin	SimpleBLE	Sensors	RAK12003_FIR_MLX90632		
Quit	Ctrl+Q	SPI	Solutions	RAK12500 GPS ZOE-M8Q >		
/ Might need	adjustments	SPIFFS	>	RAK15000 EEPROM AT24C02		
define SEALE	VELPRESSURE_H	Ticker	>	RAK15001_Flash_GD25Q16C		
		Update	>	RAK18000 Stereo		
01d Dme680_1	nit()	WebServer	>	RAK18001 Buzzer		
Wire.begin();	WiFi	>	- RAK19002 Boost TPS61046		
		WiFiClientSecure	>	RAK1901 Temperature Humidity SHTC3		
if (!bme.be	gin(0x76)) {	WiFiProv		RAK1902 Pressure LPS22HB		
return;	Incin(courd		· ·	RAK1903 Optical OPT3001		
}		Examples from Custom Libraries		RAK1904 Accelerate LIS3DH		
		Adafruit Circuit Playground	>	RAK1906 Environment BME680		
// Set up o	eratureOversa	Adafruit NeoPixel	>	RAK1910 GPS UBLOX7		
bme.setHumi	dityOversampl	ESP8266 and ESP32 OLED driver for SSD1306 displays		RAK1920 Grove Color TCS3472		
bme.setPres	sureOversampl	Heltec ESP32 Dev-Boards	>	RAK1920 Grove PIR AS312		
bme.setIIRF	ilterSize(BME	NimBLE-Arduino		RAK1920 MikroBUS Temperature TMP102		~
		SparkFun ATECCX08a Arduino Library		RAK1920 OWIIC AirOuality SGP30		
		TTGO TWatch Library		is accedented and an and an an		
		∇				

Figure 10: Opening RAK1906 example code for RAK11200 WisBlock Core

Blink Arc	duino 1.8.15 etch Tools Help						- 0	×
New	Ctrl+N							
Open	Ctrl+0							22
Open R	lecent 2							
Sketchb	book	•						^
Example	les 3	▲ (1)						
Close	Ctrl+W	06.Sensors	> d. repeatedly			1		
Save	Ctrl+S	07.Display	> repeatedly		RAK12002_RTC_Alarm_RV-3028-C7			
Save As	Ctrl+Shift+S	08.Strings	>1 the UNO, MEG	A and ZERO	RAK12002_RTC_DateTime_RV-3028-C7			
		09.USB	> LED_BUILTIN	is set to	RAK12002_RTC_EEPROM_RV-3028-C7			
Page Se	etup Ctrl+Shift+P	10.StarterKit_BasicKit	> cted to on vo	ur Arduino	RAK12003_FIR_MLX90632			
Print	Ctrl+P	11.ArduinoISP	>		RAK12010_VEML7700_Light			
Preferer	nces Ctrl+Comma	Examples for any board			RAK12011_MEMS_Pressure_LPS33HW			
		Pridge	>		RAK12500_GPS_ZOE-M8Q >			
Quit	Ctrl+Q	Ethernet	\$		RAK15000_EEPROM_AT24C02			
15 modi:	fied 2 Sep 2016	Firmata	\$		RAK15001_Flash_GD25Q16C			
16 by A:	rturo Guadalupi	LiquidCoustal	\$		RAK18001_Buzzer			
18 by C	olby Newman	sp	S.		RAK19002_Boost_TPS61046			
19	-	Stopper	5		RAK1901_Temperature_Humidity_SHTC3			
20 This	example code is	Tambaa			RAK1902_Pressure_LPS22HB			
21 22 http:	e://wase arduino c	PETIPED) nk		RAK1903_Optical_OPT3001			
23 */	5.77 WWW. al au lio. c	RETIRED			RAK1904_Accelerate_LIS3DH			
24		Examples for WisBlock RAK11300			RAK1906_Environment_BME680			
25 // the	setup function r	PDM	> power the boa	rd	RAK1910_GPS_UBLOX7			
20 Void St 27 // ii	etup() { nitialize digital	RAK WisBlock examples	RAK11300	Communications	RAK1920_Grove_Color_TCS3472			
28 pinMe	ode (LED_BUILTIN,	Scheduler	>	10 >	RAK1920_Grove_PIR_AS312			
29 }		ThreadDebug	>	Power >	RAK1920_MikroBUS_Temperature_TMP102			
30 31 // the	loop function ru	Examples from Custom Libraries	-	Sensors >	RAK1920_QWIIC_AirQuality_SGP30			
or // one	Toop function fo	Adafruit BME690 Library	.	Solutions >			_	~
1						WisBI	ock RAK11300	on COM8

Figure 11: Opening RAK1906 example code for RAK11310 WisBlock Core

3. Once the example code is open, install the Adafruit BME680 [27] library by clicking the yellow highlighted link, as shown in Figure 12 and Figure 13.

<pre>introduction with the interval int</pre>	PAV1006 Emvironment BME6001 Arduine 1.9.13	- 7	×
<pre>RV:1906_Environment_EME650 RV:1906_Environment_EME650 RV:1906_Environment_EME650.inc % Stile RV:1906_Environment_EME650 % Stile RV:1906_Environment % Stile RV:1</pre>	File Edit Sketch Tools Help		~
<pre>/** /** /** /** /** /** /** /** /** /**</pre>			Ø
<pre>/** file RAK1966_Environment_BEM600.ino fauthor rakireless.com Derief Setup and read values from bme680 menoor Pereion 0.1 deate 2020-12-28 foodback dire.by finclude ddafrui_BME600.bx // Click to install library: http://librarymanager/All&Adafruit_BME680 Adafruit_BME600 bme; // Whigh teed adaptustes foodback distants foodback dis</pre>	RAK1906 Environment BME680		
<pre>#include <d&ire.h> #include #includ</d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></d&ire.h></pre>	/** file RAK1906_Environment_BEM680.ino @author rakwireless.com Bbrief Setup and read values from bme680 sensor @version 0.1 @date 2020-12-28 @copyright Copyright (c) 2020 **/		
<pre>Adafruit_EME680 bme; // Might need adjustments #define SEALEVELPRESSURE_HPA (1010.0) void bme680_init() { Wire.begin(); if ('bme.begin(0x76)) { Serial.println("Could not find a valid BME680 sensor, check wiring!"); return; } // Set up oversampling and filter initialization bme.setTemperatureOversampling(BME680_OS_EX); bme.setHumidityOversampling(BME680_SS_X); bme.setHumidityOversampling(BME680_SS_X); bme.setIIRFilterSize(BME680_FILTER_SIZE_3);</pre>	<pre>#include <wire.h> #include <adafruit_sensor.h> #include <adafruit_bme680.h> // Click to install library: http://librarymanager/All#Adafruit_BME680</adafruit_bme680.h></adafruit_sensor.h></wire.h></pre>		
<pre>void bme680_init() { Wire.begin(); if (!bme.begin(0x76)) { Serial.println("Could not find a valid BME680 sensor, check wiring!"); return; } // Set up oversampling and filter initialization bme.setTemperatureOversampling(BME680_0S_EX); bme.setDressureOversampling(BME680_0S_EX); bme.setPressureOversampling(BME680_0S_EX); bme.setIRFilterSize(BME680_FILTER_SIZE_3); </pre>	Adafruit_BME680 bme; // Might need adjustments #define SEALEVELPRESSURE_HPA (1010.0)		
<pre>if (!bme.begin(0x76)) { Serial.println("Could not find a valid BME680 sensor, check wiring!"); return; } // Set up oversampling and filter initialization bme.setTemperatureOversampling(BME680_OS_2X); bme.setHunidityOversampling(BME680_OS_2X); bme.setFressureOversampling(BME680_OS_4X); bme.setTIRFilterSize(BME680_FILTER_SIZE_3); </pre>	<pre>void bme680_init() { Wire.begin();</pre>		
<pre>// Set up oversampling and filter initialization bme.setTemperatureOversampling(BME680_OS_8X); bme.setHumidityOversampling(BME680_OS_2X); bme.setPressureOversampling(BME680_OS_4X); bme.setTIRFilterSize(BME680_FILTER_SIZE_3); </pre>	<pre>if (!bme.begin(0x76)) { Serial.println("Could not find a valid BME680 sensor, check wiring!"); return; }</pre>		
	<pre>// Set up oversampling and filter initialization bme.setTemperatureOversampling(BME680_05_9X); bme.setRumidityOversampling(BME680_05_4X); bme.setIIRFilterSureOversampling(BME680_05_4X); bme.setIIRFilterSize(BME680_FILTER_SIZE_3);</pre>		,

Figure 12: Select Adafruit BME680 library

💿 Lik	orary Manager					×
Туре	All	√ To	opic All	~	Adafruit_BME680	
Ada	fruit BME680	Librar	У			^
by Ard	Adafruit Juino library f	for BM	E680 a	nd BME688 sensors	s. Arduino library for BME680 and BME688 humidity and pressure sensors.	
Mor	<u>re info</u>				Version 2.0.1 V	
					Close	e

Figure 13: Installing Adafruit BME680 library

During the installation, you will be asked to install the **Adafruit Unified Sensor** library. Install it as well.

🤓 Library Manager	×					
Type All V Topic All V Adafruit_BME680						
Adafruit BME680 Library	^					
/ Adafruit rduino library for BME680 and BME688 sensors. Arduino library for BME680 and BME688 humidity and pressure sensors.						
More info Dependencies for library Adafruit BME680 Library:2.0.1 X						
The library Adafruit BME680 Library:2.0.1 needs some other library dependencies currently not installed:	Install					
- Adafruit GFX Library						
Would you like to install also all the missing dependencies?						
Install all Install 'Adafruit BME680 Library' only Cancel						
	v					
	Close					

4. After successful installation of the library, you can now select the right serial port and upload the code, as shown in **Figure 15** and **Figure 16**.

Environment_Mor File Edit Sketch Tor	itoring Arduino 1.8.13			- 0	×		
	Auto Format	Ctrl+T					
	Archive Sketch						
Environment_N	Fix Encoding & Reload						
/**	Manage Libraries	Ctrl+Shift+I			^		
Gfile Envi	Serial Monitor	Ctrl+Shift+M					
@brief Thi	Serial Plotter	Ctrl+Shift+L from	BME680				
and sen @version 0	WiFi101 / WiFiNINA Firmware Updater						
@date 2020	Board: "WisBlock Core RAK4631 Board"	>					
**/	Bootloader: "0.3.2 SoftDevice s140 6.1.1"	· · · · · ·					
	Debug: "Level 0 (Release)"	>					
<pre>#include <ard #include="" <lop<="" pre=""></ard></pre>	Port	> Se	rial ports				
#include <spi< td=""><td>Get Board Info</td><td>C</td><td>DM12</td><td></td><td></td></spi<>	Get Board Info	C	DM12				
#include <₩ir	Programmer	>					
#include <ada< td=""><td>Burn Bootloader</td><td></td><td></td><td></td><td></td></ada<>	Burn Bootloader						
<pre>#include <adafru< pre=""></adafru<></pre>	include < Adafruit_EME 680.h> // Click to install library: <u>http://librarymanager/All#Adafruit_BME680</u>						
#include <u8g211< td=""><td colspan="7">nclude <u8g2lib.h> // Click to install library: <u>http://librarymanager/ALL#u8g2</u></u8g2lib.h></td></u8g211<>	nclude <u8g2lib.h> // Click to install library: <u>http://librarymanager/ALL#u8g2</u></u8g2lib.h>						
U8G2_SSD1306_128	J862_SSD1306_128X64_NONAME F HW_I2C_u8g2(U862_R0);						
Adafruit_BME680	Adafruit_BME680 bme;						
// RAK4630 suppl	// BX4630 supply two LFD						
#ifndef LED_BUIL	TIN						
#define LED_BUIL	TIN 35						
#end11	fendir						
#ifndef LED_BUIL	TIN2						
<pre>#define LED_BUIL fendif</pre>	TIN2 36						
#endir					~		



© Environment_Monitoring Arduino 1.8.13	٥	×
		ø
Environment_Monitoring		
<pre>%* %file Environment_Monitoring.ino %author rakwireless.com %brief This sketch demonstrate how to get environment data from BME680 and send the data to lora gateway. %version 0.1 %date 2020-07-28 %enversion Committed Committee Comm</pre>		
scopyright copyright (c) 2020		
<pre>#include <arduino.h> #include <lorawan-rak4630.h> // Click to install library: <u>http://librarymanager/ALL#SX126x-Arduino</u> #include <spi.h></spi.h></lorawan-rak4630.h></arduino.h></pre>		
<pre>#include <wire.h> #include <adafruit_sensor.h> #include <adafruit_bme680.h> // Click to install library: <u>http://librarymanager/All#Adafruit_BME680</u> #include <u8g2lib.h> // Click to install library: <u>http://librarymanager/All#Adafruit_BME680</u></u8g2lib.h></adafruit_bme680.h></adafruit_sensor.h></wire.h></pre>		
U8G2_SSD1306_128X64_NONAME_F_HW_I2C u8g2(U8G2_R0); Adafruit_BME680 bme;		
// RRK4630 supply two LED #ifndef LED_BUILTIN #define LED_BUILTIN 35 #endif		
<pre>\$ifndef LED_BUILTIN2 \$define LED_BUILTIN2 36 \$endif</pre>		~
1 WitBlock Core RAK4031 Board, 0.3.2 SoftDevice s140 0.1.1. Level 0 (Reid	ease) on Q	OM6

Figure 16: Uploading the RAK1906 example code

The RAK1906_Environment_BME680 application gets BME680 sensor data every 5 seconds and print the results on serial port.

Temperature = 30.66 *C	
Pressure = 1013.80 hPa	
Humidity = 44.38 %	
Gas = 3.89 KOhms	
Temperature = 30.69 *C	
Pressure = 1013.78 hPa	
Humidity = 44.59 %	
Gas = 3.88 KOhms	
Temperature = 30.72 *C	
Pressure = 1013.82 hPa	
Humidity = 44.37 %	
Gas = 3.87 KOhms	

LoRaWAN Environment Monitoring with RAK1906

This solution shows how to create an environmental data sensor node. The sensor node measures air temperature, air humidity, barometric pressure, and gas levels. After the node joins to server successfully, it then transmits this data periodically to a LoRaWan® server.

These are the quick links that go directly to the specific WisBlock Core module solution that you can use in your RAK1906 project.

- RAK1906 solution for RAK4631 12
- RAK1906 solution for RAK11310

RAK4631 Solution Requirements

- WisBlock Base RAK5005-O * 1pcs
- WisBlock Core RAK4631 * 1pcs
- WisBlock Sensor RAK1906 * 1pcs
- WisBlock Display RAK1921 * 1pcs
- LoRa Gateway * 1pcs
- Arduino IDE IDE IDE
- RAK4630 BSP 🗹
- Adafruit BME680 Library ⊡
- SX126x-Arduino Library I
- U8g2 library
 [™]
 [™]

RAK11310 Solution Requirements

- WisBlock Base RAK5005-O * 1pcs
- WisBlock Core RAK11310 * 1pcs
- WisBlock Sensor RAK1906 * 1pcs
- WisBlock Display RAK1921 * 1pcs
- LoRa Gateway * 1pcs
- Arduino IDE ☑
- Arduino BSP for RAK11300 module
- Adafruit BME680 Library
- U8g2 library
 [™]
 [™]

Configuring LoRaWAN

There are configurations that you need to setup to ensure that the device can join a LoRaWAN Network server.

The steps below will explain the default settings and how to configure it.

1. Setup the LoRa Region.

Default region is EU868.

LoRaMacRegion_t g_CurrentRegion = LORAMAC_REGION_EU868;

You can change this to a region that is applicable to you like LORAMAC_REGION_US915 , LORAMAC_REGION_AU915 , etc.

2. Setup the LoRaWAN activation method.

Default is **OTAA**.

bool doOTAA = true;

To configure the device to ABP, you need to make this boolean variable false .

3. Setup the message type if confirmed or not.

Default is confirmed message.

lmh_confirm g_CurrentConfirm = LMH_CONFIRMED_MSG;

You can change to unconfirmed message by changing the value to LMH_UNCONFIRMED_MSG .

4. Setup device class.

Default is Class A.

DeviceClass_t g_CurrentClass = CLASS_A;

You can change this to CLASS_B (still under development) or CLASS_C.

5. Setup the keys.

Configuration keys for OTAA Activation:

```
uint8_t nodeDeviceEUI[8] = {0x88, 0x88, 0x88, 0x88, 0x88, 0x88, 0x33, 0x33;;
uint8_t nodeAppEUI[8] = {0x88, 0x27, 0xEB, 0xFF, 0xFE, 0x39, 0x00, 0x00};
uint8_t nodeAppKey[16] = {0x88, 0x88, 0x88
```

Configuration keys for ABP Activation:



6. Setup uplink period.

The default is **20000 mS**.

#define LORAWAN_APP_INTERVAL 20000

Data Format

Byte1	Byte2	Byte3	Byte3		Byte4		Byte5	
1	Temperature1	Tempe	Temperature2		Humidity1			
Byte6	Byte7	Byte8	Byte9	Byte10	Byte11	Byte12	Byte13	
Barometric1	Barometric2	Barometric3	Barometric4	Gas1	Gas2	Gas3	Gas4	

- Byte1 is a marker for the data type, here always 1.
- Every byte value is MSB first.
- Temperature is sent as two bytes, enlarged 100 times for accuracy. For example, 2510 means 25.01 C, sent as 0x09 0xCE.
- Humidity is sent as two bytes, enlarged 100 times for accuracy. For example, 4173 means 41.73%, sent as 0x10 0x4D.
- Barometric pressure is sent as four bytes, enlarged 100 times for accuracy. For example, 100945 means 1009.45 hPa, sent as 0x00 0x01 0x8A 0x51.
- Gas resistance is sent as four bytes. For example, 63560 Ohm, sent as 0x00 0x00 0xF8 0x48.

Attention

If this example is implemented for the Region US915, DR0 cannot be used because the package size is too large to fit into the allowed payload.

When you successfully uploaded the example sketch, open the Serial Monitor of the Arduino IDE to see the sensor's reading logs. If you see the logs, as shown below, then your RAK1906 is properly communicating to the WisBlock core.

BAK[®] Documentation Center

10:30:07.777	->		
10:30:07.777	->	Welcome to RAK4630 LoRaWan!!!	
10:30:07.777	->	Туре: ОТАА	
10:30:07.777	->	Region: EU868	
10:30:07.777	->		
10:30:07.777	->	Chip ID=0×61	
10:30:12.974	->	OTAA Mode, Network Joined!	
10:30:33.999	->	Sending frame now	
10:30:33.999	->	result: Tem:28.70C Hum:67.00% Pres:948.50KPa Gas:276460hms	
10:30:33.999	->	lmh_send ok count 1	
10:30:53.979	->	Sending frame now	
10:30:53.979	->	result: Tem:28.70C Hum:67.00% Pres:948.50KPa Gas:2424880hms	
10:30:54.012	->	lmh_send ok count 2	
10:31:14.006	->	Sending frame now	
10:31:14.006	->	result: Tem:28.70C Hum:67.00% Pres:948.50KPa Gas:4571740hms	
10:31:14.006	->	lmh_send ok count 3	
10:31:33.997	->	Sending frame now	
10:31:33.997	->	result: Tem:28.69C Hum:67.00% Pres:948.50KPa Gas:741020hms	
10:31:34.032	->	lmh_send ok count 4	

Data arrives at LoRaWAN server.





Last Updated: 1/10/2022, 2:07:11 AM