

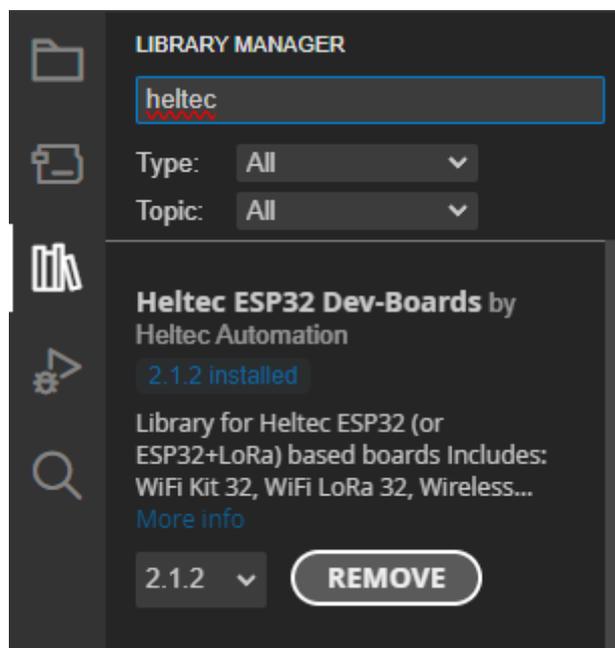
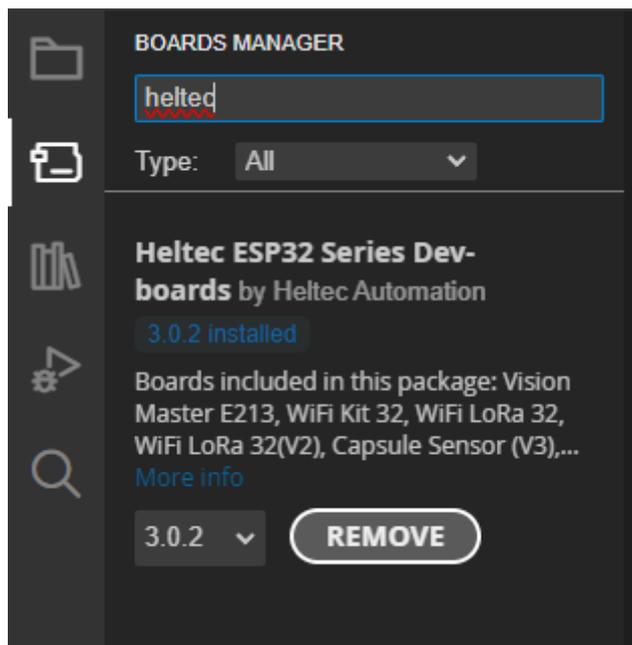
# Usar LORA na IDE 2.3.4

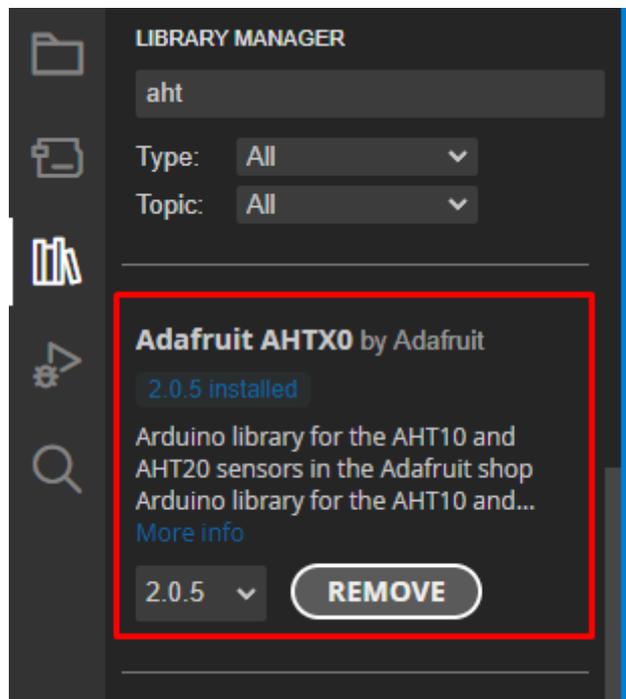
## Configuração do ambiente

Adicionar link a seguir nas preferências da IDE.

[https://resource.heltec.cn/download/package\\_heltec\\_esp32\\_index.json](https://resource.heltec.cn/download/package_heltec_esp32_index.json)

Instalar o suporte à **placa** e **biblioteca**.





E uma biblioteca externa para o **ENSx**.

<https://github.com/sciosense/ens16x-arduino/tree/main>

## Ativando o rádio

Para ligar o rádio, agora precisa usar um comando, e esse comando faz uma verificação de código da placa.

```
1 | Mcu.begin(HELTEC_BOARD, SLOW_CLK_TPYE);
```

```
void setup() {  
  Serial.begin(115200);  
  Serial.println();  
  Serial.println();  
  VextON();  
  delay(100);  
  
  // Initialising the UI will init the display too.  
  display.init();  
  display.display();  
  
  display.setContrast(255);  
  
  display.setFont(ArialMT_Plain_10);  
  
  Mcu.begin(HELTEC_BOARD, SLOW_CLK_TPYE);  
  
  txNumber = 0;  
}
```

Quando fizer isso, **se a placa não estiver registrada**, você precisa registrar. Para saber disso, depois de compilar o código, na serial começará aparecer uma mensagem requisitando o registro.

A mensagem será seguida de um código individual da placa.

Abra o site: <https://resource.heltec.cn/search>

Please enter product id:

D4AD0312CFA4

Confirm

Reset

Cole o código e depois confirme.

Será aberto a página que mostra o código de ativação:

Type	id	license	Production Place	Manufacture Data
	D4AD0312CFA4	0xD462EB69,0xF6C2E984,0xBF431B1A,0x0079F54F		

Back

Relevant Resource

Copie esse código (cada placa tem um código).

Agora, na serial digite:

```
AT+CDKEY=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

Onde xxxxxxxx é código copiado do site. Mas preste atenção: Remova todos os **0x** e **,**

Exemplo: AT+CDKEY=D462EB69F6C2E.....

Após isso, será mostrado uma mensagem de ativação com sucesso na serial.

Veja mais detalhes: [https://docs.heltec.org/general/how\\_to\\_use\\_license.html](https://docs.heltec.org/general/how_to_use_license.html)

## Pacote de dados

Os dados são enviados em formato inteiro (multiplicando por 100 para manter 2 casas decimais) e no receptor podemos voltar para float, dividindo por 100. O pacote completo possui 9 bytes

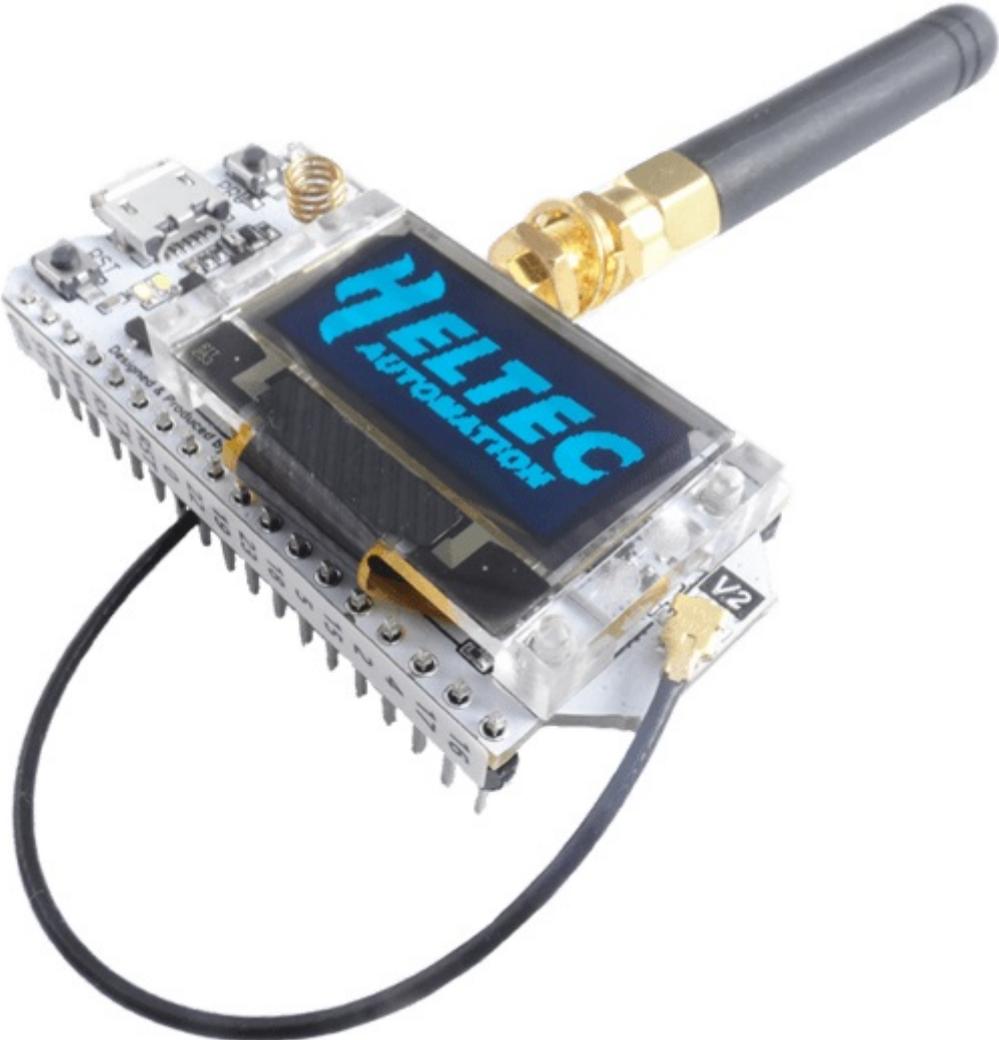
1 uint8\_t = 1 byte = Qualidade do ar (pode variar de 1 a 100)

1 uint32\_t = 4 bytes = Temperatura (pode variar de 1 a 100)

1 uint32\_t = 4 bytes = Umidade (pode variar de 1 a 100)

# Sender

---



Auto Format	Ctrl+T
Archive Sketch	
Manage Libraries...	Ctrl+Shift+I
Serial Monitor	Ctrl+Shift+M
Serial Plotter	
Firmware Updater	
Upload SSL Root Certificates	
Board: "WiFi LoRa 32(V2)"	▶
Port: "COM39"	▶
Reload Board Data	
Get Board Info	
CPU Frequency: "240MHz (WiFi/BT)"	▶
Core Debug Level: "None"	▶
Erase All Flash Before Sketch Upload: "Disabled"	▶
LoRaWan DevEUI: "CUSTOM"	▶
LoRaWan Preamble Length: "8(default)"	▶
LoRaWan Region: "REGION_AU915"	▶
LoRaWan Debug Level: "None"	▶
Upload Speed: "921600"	▶
Programmer	▶
Burn Bootloader	

## Receiver



Tools	Help
Auto Format	Ctrl+T
Archive Sketch	
Manage Libraries...	Ctrl+Shift+I
Serial Monitor	Ctrl+Shift+M
Serial Plotter	
<hr/>	
Firmware Updater	
Upload SSL Root Certificates	
<hr/>	
Board: "Vision Master T190"	▶
Port: "COM83"	▶
Reload Board Data	
Get Board Info	
<hr/>	
USB CDC On Boot: "Enabled"	▶
CPU Frequency: "240MHz (WiFi)"	▶
Core Debug Level: "None"	▶
USB DFU On Boot: "Disabled"	▶
Erase All Flash Before Sketch Upload: "Disabled"	▶
Events Run On: "Core 1"	▶
Arduino Runs On: "Core 1"	▶
LoRaWan DevEUI: "CUSTOM"	▶
LoRaWan Preamble Length: "8(default)"	▶
LoRaWan Region: "REGION_EU868"	▶
LoRaWan Debug Level: "None"	▶
USB Firmware MSC On Boot: "Disabled"	▶
Slow Clk Type(only for LoRaWAN): "Internal (default)"	▶
Upload Mode: "UART0 / Hardware CDC"	▶
Upload Speed: "921600"	▶
USB Mode: "Hardware CDC and JTAG"	▶
<hr/>	
Programmer	▶
Burn Bootloader	