

LoRaSDL

Sensor distancia, temperatura y humedad,
tecnología LoRa



Modelos disponibles:

- LoRaSDL-S: DISTANCIA (*)
- LoRaSDL-TH: TEMPERATURA Y HUMEDAD
- LoRaSDL-STH: DISTANCIA, TEMPERATURA Y HUMEDAD

Documentación, esquemas, librerías, ejemplos, etc.:



<https://github.com/raymirabel/LoRaSDL>

ÍNDICE DE CONTENIDO

1. Descripción general.....	3
2. Características principales.	4
3. Microcontrolador.....	5
4. Programación.....	6
5. Línea de comandos.....	9
6. Especificaciones técnicas generales.	13
7. Especificaciones sensor distancia.....	14
8. Especificaciones sensor temperatura y humedad.	14
9. Revisiones.	15

1. Descripción general.

El sensor LoRaSDL es un dispositivo con tecnología LoRa que permite monitorizar parámetros como distancia, temperatura, humedad y luz ambiente en entornos interiores y exteriores a través de las redes IoT (Internet de las Cosas).

LoRaSDL incorpora un microcontrolador ATmega328P a 16Mhz y el diseño de la electrónica está basado en arquitectura **Arduino™ UNO** lo cual nos permite fácilmente modificar el firmware en cualquiera de los entornos de programación compatibles con esta plataforma.

Pueden combinarse, bajo pedido, cualquiera de los distintos sensores disponibles:

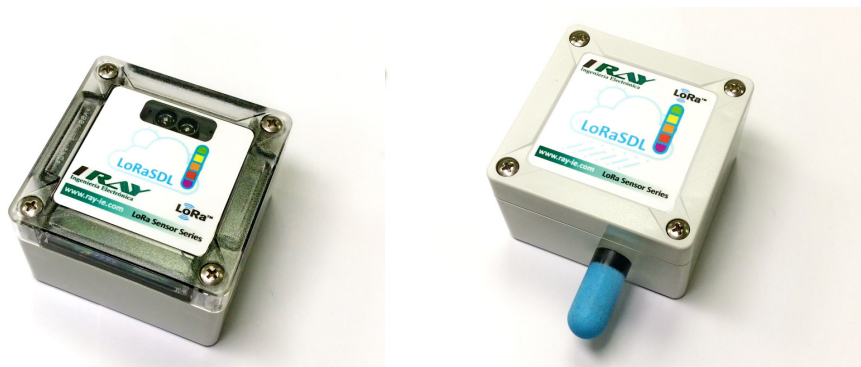
- **Distancia (modelo estándar o base).**
- Temperatura y humedad.
- Distancia, temperatura y humedad.
- Sensor LDR o luz ambiente integrado en todos los modelos

El sensor se entrega montado en una caja discreta y robusta de policarbonato con protección IP67.

Está recomendado para aplicaciones domóticas, monitorización de climatización y calidad del aire en edificios, eficiencia energética, medida nivel depósitos, etc.

2. Características principales.

- Microcontrolador Atmel ATmega328P compatible con **Arduino Uno**.
- Compatible LoRa/LoRaWAN 868Mhz (RFM95).
- 4 interruptores dipswitch propósito general o dirección de nodo.
- Alimentación pilas 2xAA.
- Regulador conmutado de alta eficiencia y bajo consumo.
- Medidas disponibles: distancia, temperatura, humedad y luz ambiente.
- Tamaño compacto.
- Bajo consumo en modo sleep <40uA.



LoRaSDL-S y LoRaSDL-TH

3. Microcontrolador.

LoRaSDL se entrega con un firmware ejemplo estándar de funcionamiento. Dicho firmware transmite paquetes LoRa cada cierto tiempo con una estructura de datos donde se encuentran los parámetros de las medidas de distancia, temperatura, humedad, luz, nivel de batería, etc. Para más información, consulte el apartado 5 sobre la línea de comandos.

Hay disponible una librería y ejemplos para facilitar la programación si desea modificar o mejorar el firmware actual. Puede encontrar el repositorio de librerías, documentación, manuales, etc, en este enlace:

<https://github.com/raymirabel/LoRaSDL.git>

El dispositivo está basado en Arduino y por consiguiente podremos utilizar cualquiera de los entornos de desarrollo Arduino para modificar el firmware.

La equivalencia entre las E/S de LoRaSDL y Arduino son las siguientes:

E/S LoRaSDL	E/S Arduino
UART RX	0
UART TX	1
GPIO0 - IRQ RFM95	2
DS18B20	3
CS RFM95	4
GPIO1 - RFM95	5
RX LIDAR	6
TX LIDAR	7
ENABLE SENSOR	8
LED TEST	9
ENABLE LIDAR	10
DIPSWITCH 1	A0
DIPSWITCH 2	A1
DIPSWITCH 3	A2
DIPSWITCH 4	A3
DS18B20 DATA	A4
I2C SDA	A4
I2C SCL	A5
VBAT	A6
LDR	A7

4. Programación.

(*) **NOTA IMPORTANTE:** Asegúrese que tiene seleccionado, dentro del entorno de desarrollo, como placa el modelo "Arduino UNO".

Disponemos de dos alternativas para programar o modificar el firmware de nuestro sensor LoRaSDL:

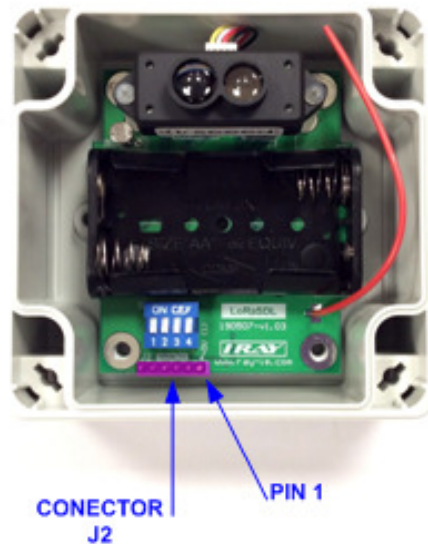
- Mediante el puerto serie TTL disponible y el bootloader cargado en el microcontrolador. El sensor LoRaSDL se entrega con el bootloader de **Arduino UNO** cargado por lo que solo necesitamos un puerto serie TTL en nuestro ordenador para descargar programas desde el IDE de Arduino.

Existen cables USB comerciales con salida serie TTL. Nosotros recomendamos, el modelo TL-232R-5V de la firma FTDI:



El pinout del conector J2 es el siguiente:

PIN	SEÑAL	COLOR
1	GND	NEGRO
2	CTS	MARRON
3	NO UTILIZADO	ROJO
4	TXD (RX del microcontrolador)	NARANJA
5	RXD (TX del microcontrolador)	AMARILLO
6	RTS (RESET del microcontrolador)	VERDE

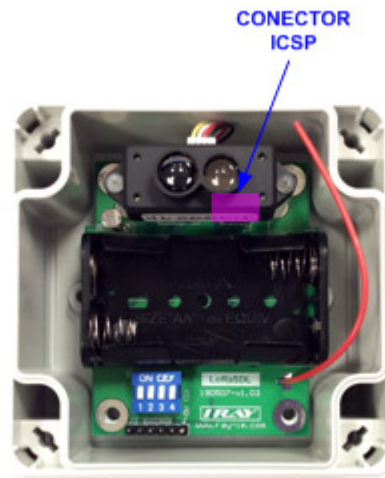


(*) **NOTA IMPORTANTE:** Antes de conectar el cable de programación, asegúrese que tiene bien posicionado el conector: PIN1 = GND = CABLE COLOR NEGRO.

- Mediante un programador externo a través del conector ICSP. Esta es la forma nativa de programar los microcontroladores AVR de Atmel. El programador más utilizado para este fin es el AVR MKII:



Para programar el microcontrolador con este programador tenemos que acceder a la placa quitando la tapa superior del sensor:



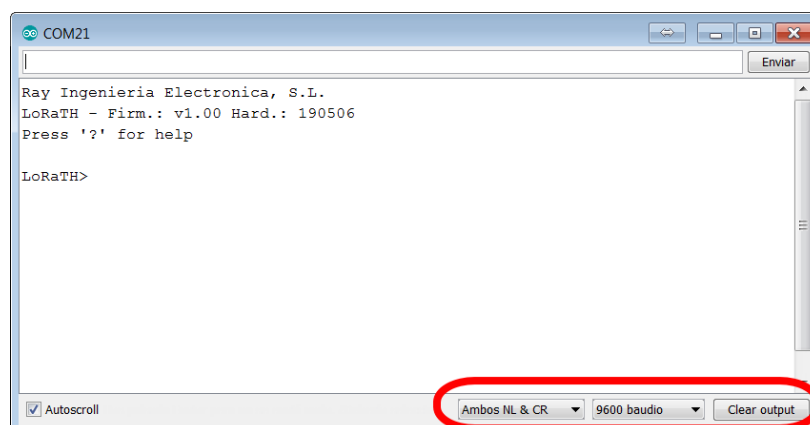
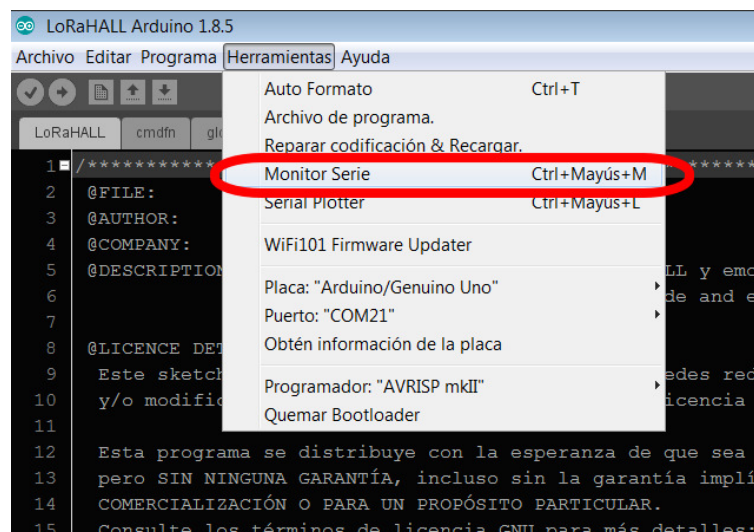
(*) NOTA IMPORTANTE: Antes de conectar el cable de programación, asegúrese que tiene bien posicionado el conector: PIN1 = CABLE MARCADO CON BANDA ROJA.

5. Línea de comandos.

LoRaSDL se entrega con un firmware ejemplo de estándar de funcionamiento llamado "**LoRa NODE RAW**". Este ejemplo transmite paquetes LoRa cada cierto tiempo con una estructura de datos donde se encuentran los parámetros de las medidas de distancia, temperatura, humedad, luz, nivel de batería, etc.

El firmware "**LoRa NODE RAW**" incluye una línea de comandos para configurar parámetros y consultar medidas.

Para acceder a la línea de comandos debe conectar un cable compatible SERIE-TTL al conector J2 de LoRaSDL y configurar el terminal serie a 9600,N,8,1. Consulte el apartado 5, programación, para más información de cómo conectar un cable serie TTL. Puede usar como terminal serie el que incluye el IDE de Arduino:



Al iniciar el terminal serie aparecerá el siguiente mensaje de bienvenida donde nos indica la versión del firmware y hardware:

```
Ray Ingenieria Electronica, S.L.  
LoRaSDL - Firm.: v1.00 Hard.: 190507  
Press '?' for help
```

Pulse ? para consultar la lista de comandos:

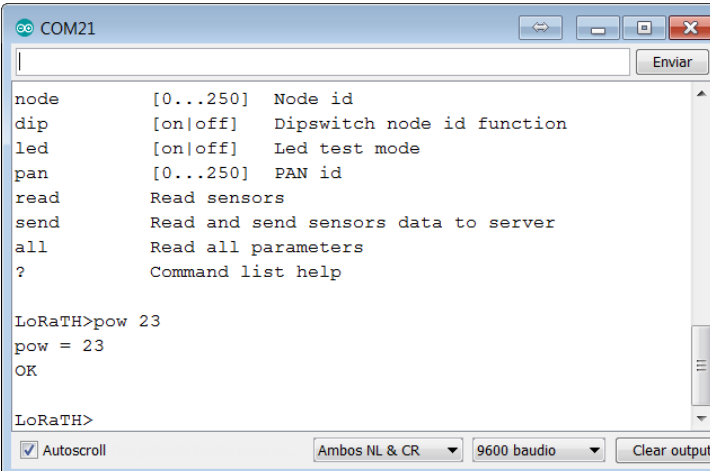
pow	[5...23]	RF power (5:min...23:max)
stime	[1...9999]	Send time delay (minutes)
retri	[0...20]	Data send retries
node	[0...250]	Node id
dip	[on off]	Dipswitch node id function
led	[on off]	Led test mode
pan	[0...250]	PAN id
fil	[0...9]	Lidar filter (0:min...10:max)
res	[1...100]	Lidar resolution (cm)
repe	[1...10]	Lidar measurement repeats
read		Read sensors
send		Read and send sensors data to server
all		Read all parameters
?		Command list help

A continuación detallamos cada comando:

- **pow:** establece la potencia de transmisión LoRa.
- **stime:** establece el periodo de envío de datos en minutos.
- **retri:** reintentos de transmisión cuando los datos no llegan al servidor LoRa.
- **node:** identifica al nodo. Puede haber hasta 250 nodos por cada PAN.
- **dip:** establece si la dirección del nodo es tomada mediante el dipswitch (dirección de 0 a 15) o mediante el comando "node".

- **led**: activa o desactiva el LED de test cada vez que el nodo despierta. Este indicador, cuando está activado, nos muestra cuando el nodo está activo leyendo los sensores y transmitiendo los datos. Podemos reducir el consumo desactivando este indicador.
- **pan**: identificador PAN o red de área personal. Solo los nodos y servidor con el mismo identificador PAN comunicarán entre sí. Este identificador es útil para separar redes dentro de un mismo área.
- **fil**: filtro medida distancia. Un valor de 0 desactiva el filtro.
- **res**: establece la resolución de medida de distancia.
- **repe**: repeticiones de medida para la distancia. A mayor número de repeticiones el tiempo de medida será mayor y por lo tanto mayor consumo, pero a cambio tendremos mejor precisión en la medida.
- **read**: lee y muestra de forma inmediata todos los valores de los sensores.
- **send**: lee y envía de forma inmediata todos los parámetros de energía.
- **all**: lee y muestra todos los valores de la lista de comandos

Para modificar un valor de un comando escriba el comando y el nuevo valor. Por ejemplo, para modificar la potencia de transmisión al máximo escriba **pow 23** (y retorno de carro):

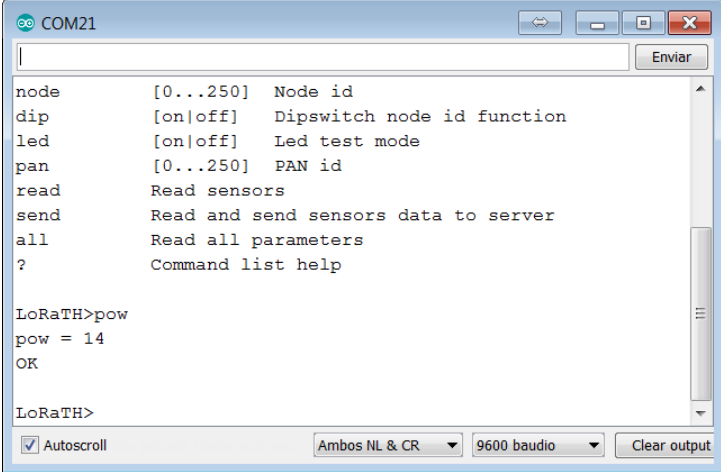


```
COM21
node      [0...250]  Node id
dip       [on|off]   Dipswitch node id function
led       [on|off]   Led test mode
pan       [0...250]  PAN id
read      Read sensors
send      Read and send sensors data to server
all       Read all parameters
?         Command list help

LoRaTH>pow 23
pow = 23
OK

LoRaTH>
```

Para consultar un valor de un comando escriba el comando y pulse retorno de carro.



The screenshot shows a serial terminal window titled "COM21". The window contains a list of commands and their descriptions:

Command	Range	Description
node	[0...250]	Node id
dip	[on off]	Dipswitch node id function
led	[on off]	Led test mode
pan	[0...250]	PAN id
read		Read sensors
send		Read and send sensors data to server
all		Read all parameters
?		Command list help

Below the list, the following interaction is shown:

```
LoRaTH>pow
pow = 14
OK
LoRaTH>
```

At the bottom of the window, there are settings: "Autoscroll" is checked, "Ambos NL & CR" is selected for line endings, "9600 baudio" is selected for baud rate, and there is a "Clear output" button.

6. Especificaciones técnicas generales.

- Alimentación: 2xAA (alcalina)
- Rango de funcionamiento: 1.5 ~ 3.3 VDC
- Consumo máximo: 150mA
- Consumo sensor off: 11mA
- Consumo modo sleep: 40uA
- Microcontrolador: Atmega328P @ 16Mhz
- Memoria Flash: 32K
- Memoria RAM: 2Kb
- Transceiver LoRa: RFM95
- Temperatura de funcionamiento: -10 ~ 60 °C
- Grado de protección IP: IP67
- Ancho: 85 mm
- Alto: 85 mm
- Fondo: 55 mm
- Peso (pilas inc.): 230 g.

7. Especificaciones sensor distancia.

- Modelo: LoRaSDL-S
- Tipo sensor: TF Mini LiDAR (SEED STUDIO)
- Interface: UART TTL
- Resolución: 0.5 cm
- Rango de medida: 0.3m ~ 12m
- Precisión de 0.3 a 6m: +/- 4cm
- Precisión de 6 a 12m: +/- 6cm

8. Especificaciones sensor temperatura y humedad.

- Modelo: LoRaSDL-TH
- Tipo sensor: SHT21 (SENSIRION)
- Interface: I2C
- Resolución temperatura: 0.1°C
- Rango de medida: -40 ~ 85°C
- Precisión típica: +/- 0.3°C
- Precisión máxima: +/- 1°C
- Resolución humedad: 1%
- Rango de medida: 0 ~ 100%RH
- Precisión típica: +/- 2%RH
- Precisión máxima: +/- 5%RH

9. Revisiones.

FECHA	VERSION	NOTAS
20/07/2019	1.00	- Documento inicial.