

---

# IoTOS

*Internet of Things OS*

## Station Météo — Récepteur LoRa

---

**Documentation Technique Complète**

**Arduino UNO Rev 4 Minima • Ethernet Shield V2 (W5500)**

Grove Base Shield • LoRa 868 MHz • LCD ST7567 128×64

Serveur WEB AJAX • Shell CLI Serial / Telnet • Datalogger SD

**Biechy Jean-Marc**

École d'Informatique — Institution Saint-Jean Colmar

Version 73 — 2026

# 1. Introduction

IoToS (Internet of Things OS) est un micro-système d'exploitation embarqué pour Arduino UNO R4 Minima. Il transforme la carte Arduino en une station météo connectée capable de recevoir des données par radio LoRa 868 MHz, de les diffuser sur un serveur WEB local, de les enregistrer sur carte SD et de les afficher sur un écran LCD.

Ce projet a été développé à l'École d'Informatique de l'Institution Saint-Jean de Colmar par Biechy Jean-Marc. Il est conçu pour usage pédagogique et illustre l'intégration de multiples technologies IoT sur une plateforme Arduino.

## 1.1 Résumé du Système

Le système est composé de deux unités distinctes communiquant par radio :

Unité Émettrice (Arduino UNO R3)	Unité Réceptrice IoToS (Arduino UNO R4)
Grove Shield + capteurs physiques	Arduino UNO R4 Minima + Ethernet Shield V2
Anémomètre RJ11 (D) → broche D2	Récepteur LoRa 868 MHz → Serial1 D0/D1
Girouette RJ11 (D) → broche A0	Serveur WEB AJAX (port 80) sur carte SD
Pluviomètre RJ11 (G) → broche D4	Shell CLI Serial + Telnet (port 23)
DHT22 Température/Humidité → D5	Écran LCD I2C 128x64 ST7567
Émission LoRa 868 MHz D6/D7	Datalogger CSV sur Micro SD
Trame : STX   ID   LEN   DATA   CRC   ETX	API météo externe Open-Meteo (HTTP)

## 1.2 Architecture Logicielle

Le firmware est organisé en couches de fichiers .h spécialisés :

- Ethernet\_Reseau.h : initialisation réseau IP/DHCP, diagnostics
- ServeurWEB.h : serveur HTTP AJAX, routage des pages, authentification
- Shell\_CmdLine\_SD.h : CLI Série/Telnet, gestion fichiers SD
- GetMeteo.h : client HTTP Open-Meteo, parsing JSON sans ArduinoJson
- TrameStandard\_LoRa\_CRC\_Recive.h : machine à états réception LoRa + CRC
- LCD\_GM12864\_64x128.h : affichage LCD, menu rotatif 3 boutons
- utils.h : bannières Serial/Telnet, fonctions utilitaires

## 1.3 Fonctionnalités Principales

- Réception radio LoRa 868 MHz (vent, girouette, pluie, température, humidité, pression)
- Serveur WEB AJAX temps réel (1 s LoRa / 30 s Open-Meteo)
- Double source météo : capteurs locaux LoRa + météo Internet Open-Meteo
- Shell CLI accessible en Serial USB et en Telnet réseau
- Datalogger : enregistrement CSV horodaté NTP sur carte Micro SD
- Serveur FTP pour accès aux fichiers de la carte SD

- 
- Écran LCD 128x64 avec menu de navigation à 3 boutons
  - EEPROM : sauvegarde persistante de toute la configuration
  - Horloge NTP synchronisée sur Internet

## 2. Documentation d'Installation

### 2.1 Matériel Requis

- Arduino UNO R4 Minima (microcontrôleur principal — Renesas RA4M1)
- Arduino Ethernet Shield V2 (puce W5500 — port SD intégré)
- Grove Base Shield (connecteurs standardisés pour modules Grove)
- Module Grove LoRa 868 MHz (RFM95 — récepteur radio)
- Écran LCD GM12864-59N VER2.0 (ST7567S — I2C 128x64 pixels)
- 3x Boutons Grove (OK sur D8, (+) sur D2, (-) sur D3)
- Module LED Grove (sur port D5 — indicateur serveur WEB)
- Carte Micro SD (formatée FAT32 — noms longs activés)
- Câble RJ45 (connexion réseau Ethernet)
- Câble USB (programmation et monitoring Serial)

### 2.2 Logiciels Requis

- Arduino IDE 2.3.7 (obligatoire — l'IDE 1.x présente un bug IP fixe Ethernet)
- Carte cible : Outils → Type de carte → Arduino UNO R4 Minima

### Bibliothèques à installer

Copier les bibliothèques dans Documents/Arduino/libraries/ :

Bibliothèque	Version	Rôle
<a href="#">Ethernet</a>	1.5.3 / 2.0.2	Shield Ethernet W5500 (intégrée IDE)
<a href="#">EEPROM</a>	—	Mémoire persistante (intégrée IDE)
<a href="#">SPI</a>	—	Bus SPI (intégrée IDE)
<a href="#">SdFat</a>	2.3.1	Gestion carte Micro SD (noms longs)
<a href="#">RH_RF95 / RadioHead</a>	Grove LoRa	Réception LoRa 868 MHz
<a href="#">NTPClient</a>	—	Synchronisation horloge Internet
<a href="#">Dns</a>	—	Résolution DNS (intégrée Ethernet)
<a href="#">U8g2</a>	2.35.30	Affichage écran LCD ST7567
<a href="#">Wire</a>	—	Bus I2C pour LCD (intégrée IDE)
<a href="#">Arduino CmdLine</a>	modifiée	Shell CLI Serial/Telnet

#### Configuration SdFat obligatoire

Dans SdFatConfig.h activer : `#define USE_LONG_FILE_NAMES 1` et `#define SDFAT_FILE_TYPE 1`

## 2.3 Branchement des Modules Grove

Empiler dans l'ordre : Arduino UNO R4 Minima → Ethernet Shield V2 → Grove Base Shield.  
Connecter ensuite les modules Grove sur les ports indiqués ci-dessous.

### Ports Numériques (Digital)

Port Grove	Connecteur	Fonction	Remarque
D0 / D1	Serial1 (RJ)	Récepteur LoRa 868 MHz (TX LoRa → D0 RX Arduino)	Ne pas utiliser pour autre chose
D2	Grove Digital	Bouton (+) menu LCD — Interruption INT	attachInterrupt compatible
D3	Grove Digital	Bouton (-) menu LCD — Interruption INT	attachInterrupt compatible
D4	Grove Digital	CS Micro SD (SPI Ethernet Shield)	Réservé — câblé sur le shield
D5	Grove Digital	Module LED Grove (indicateur serveur WEB actif)	LED allumée = serveur démarré
D6	Grove Digital	LED Réception LoRa (indicateur trame reçue)	Témoin visuel réception
D8	Grove Digital	Bouton OK — validation menu LCD — Interruption INT	buttonPinOK
D10	SPI	CS Ethernet W5500 (SPI Ethernet Shield)	Réservé — câblé sur le shield

### Ports Analogiques / I2C

Port Grove	Connecteur	Fonction	Remarque
A4 (SDA)	I2C Grove	Écran LCD ST7567 — Signal SDA	Bus I2C Software (bit-banging)
A5 (SCL)	I2C Grove	Écran LCD ST7567 — Signal SCL	Bus I2C Software (bit-banging)

### Connexions LoRa Grove 868 MHz

**D6 (Grove RX) → TX Arduino** : signal de transmission vers le module

**D7 (Grove TX) → RX LoRa** : signal de réception depuis le module

**D0/D1 (Serial1 hardware)** : communication LoRa vers l'Arduino R4

#### Note I2C LCD

L'écran LCD utilise le mode Software I2C (bit-banging). Adresse I2C : 0x3F (0x7E en 8 bits). Le mode Hardware I2C présente un bug avec le contrôleur ST7567.

### Schéma de Câblage — Vue Synthétique

Capteurs physiques	Grove Shield	UNO R3	Émetteur	LoRa 868MHz	UNO R4
Récepteur					
Anémomètre (RJ11 D)	→D2/INT1	LoRa D6/D7	→Radio	→D0/D1	Serial1
Girouette (RJ11 D)	→A0/A1			D8	— Bouton OK
Pluviomètre (RJ11 G)	→D4			D2	— Bouton (+)
DHT22	→D5			D3	— Bouton (-)
LCD ST7567 I2C				A4/A5	SDA/SCL
LED Emission				D5	LED WEB

## 2.4 Préparation de la Carte Micro SD

- Formater la carte en FAT32 (MSDOS, noms longs activés)
- Copier index.htm à la racine de la carte SD
- Copier ConfIP.htm à la racine de la carte SD
- Le fichier datalogger.csv sera créé automatiquement par le système

### Insertion avant la mise sous tension

La carte Micro SD doit être insérée AVANT la mise sous tension. Sans SD : FTP et Datalogger désactivés. Le serveur WEB reste fonctionnel.

## 2.5 Paramétrage IP — Instructions CLI

Le Shell CLI est accessible via Serial USB (9600 bauds) ou via Telnet (port 23). La configuration réseau se fait entièrement par commandes texte.

### Accès au CLI

**Via Serial USB** : Ouvrir le Moniteur Série de l'IDE Arduino (9600 bauds, NL)

**Via Telnet** : Depuis un terminal : telnet 192.168.0.10 23 (IP par défaut)

**Via PuTTY** : Décocher « Return key sends Telnet New Line » dans le menu Telnet

**Via MobaXterm** : Taper cli off pour éviter le doublement des caractères

### Commandes de Configuration IP

Commande	Syntaxe	Description
<code>ip</code>	<code>ip 192.168.0.66</code>	Définir l'adresse IP statique de l'Arduino
<code>subnet</code>	<code>subnet 255.255.255.0</code>	Définir le masque de sous-réseau
<code>gateway</code>	<code>gateway 192.168.0.1</code>	Définir la passerelle (routeur)
<code>mac</code>	<code>mac DE:AD:BE:EF:FE:B2</code>	Définir l'adresse MAC (format hex)

Commande	Syntaxe	Description
<code>dhcp</code>	<code>dhcp on / dhcp off</code>	Activer ou désactiver le mode DHCP
<code>port</code>	<code>port 80</code>	Définir le port du serveur WEB (défaut : 80)

## Commandes de Sauvegarde / Restauration

Commande	Syntaxe	Description
<code>save</code>	<code>save</code>	Sauvegarder la configuration en EEPROM
<code>reload</code>	<code>reload</code>	Recharger la configuration depuis l'EEPROM
<code>reset</code>	<code>reset</code>	Redémarrer l'Arduino avec la config sauvegardée
<code>defaults</code>	<code>defaults</code>	Restaurer la configuration par défaut (192.168.0.10)
<code>saveconf</code>	<code>saveconf &lt;nom&gt;</code>	Sauvegarder la config dans un fichier SD
<code>loadconf</code>	<code>loadconf &lt;nom&gt;</code>	Charger la config depuis un fichier SD
<code>print</code>	<code>print</code>	Afficher toute la configuration réseau actuelle
<code>version</code>	<code>version</code>	Afficher la version du firmware

## Procédure de Changement d'IP — Étape par Étape

Exemple : passer de l'IP par défaut 192.168.0.10 à 192.168.1.50 :

Étape	Commande à taper	Résultat attendu
1	<code>ip 192.168.1.50</code>	IP enregistrée en RAM (pas encore active)
2	<code>subnet 255.255.255.0</code>	Masque enregistré en RAM
3	<code>gateway 192.168.1.1</code>	Passerelle enregistrée en RAM
4	<code>save</code>	Configuration écrite en EEPROM
5	<code>reset</code>	Arduino redémarre avec la nouvelle IP

## Autres Commandes CLI Utiles

Commande	Syntaxe	Description
<code>ping</code>	<code>ping 8.8.8.8</code>	Test connectivité réseau (ping complet ICMP)
<code>netstat</code>	<code>netstat</code>	Afficher les connexions réseau actives
<code>tracert</code>	<code>tracert google.com</code>	Tracer la route réseau vers une destination
<code>dir / ls</code>	<code>dir</code>	Lister les fichiers sur la carte Micro SD
<code>type</code>	<code>type fichier.txt</code>	Afficher le contenu d'un fichier SD

Commande	Syntaxe	Description
<code>del</code>	<code>del fichier.csv</code>	Supprimer un fichier de la carte SD
<code>datalogger</code>	<code>datalogger on</code>	Activer / désactiver l'enregistrement CSV
<code>WEB</code>	<code>WEB on / WEB off</code>	Activer / désactiver le serveur WEB
<code>FTP</code>	<code>FTP on / FTP off</code>	Activer / désactiver le serveur FTP
<code>RAM</code>	<code>RAM</code>	Afficher la mémoire RAM disponible
<code>LED</code>	<code>LED on / LED off</code>	Contrôler la LED intégrée D5
<code>LoRaFreq</code>	<code>LoRaFreq 868.0</code>	Régler la fréquence radio LoRa (MHz)
<code>help / ?</code>	<code>help</code>	Afficher l'aide complète de toutes les commandes

## 2.6 Réglage de la Station Météo

Paramètres spécifiques à la station météo configurables depuis le CLI.

### Localisation Géographique pour Open-Meteo

Coordonnées GPS par défaut : Colmar (Alsace) — Latitude 48.08, Longitude 7.36

Commande	Syntaxe	Description
<code>GPS</code>	<code>GPS 48.5734 7.7521</code>	Définir latitude et longitude (ex. Strasbourg)
<code>GPSville</code>	<code>GPSville Colmar</code>	Sélectionner une ville prédéfinie

Villes prédéfinies : Strasbourg, Sélestat, Colmar, Mulhouse, Nancy, Lyon, Paris

### Configuration du Pluviomètre

Commande	Syntaxe	Description
<code>resetPluvio</code>	<code>resetPluvio</code>	Remettre à zéro le compteur de pluie
<code>infoMeteoLocal</code>	<code>infoMeteoLocal</code>	Afficher les dernières valeurs LoRa reçues
<code>infoMeteoWEB</code>	<code>infoMeteoWEB</code>	Afficher les dernières valeurs Open-Meteo

### Configuration du Datalogger

Commande	Syntaxe	Description
<code>datalogger</code>	<code>datalogger on</code>	Activer l'enregistrement CSV sur SD
<code>create</code>	<code>create meteo.csv</code>	Créer un nouveau fichier datalogger
<code>select</code>	<code>select meteo.csv</code>	Sélectionner le fichier d'enregistrement actif
<code>time</code>	<code>time 5</code>	Intervalle d'enregistrement en minutes

---

## Fréquence LoRa

La fréquence LoRa par défaut est 868.0 MHz (Europe). Elle doit être identique entre émetteur et récepteur :

```
> LoRaFreq 868.0
```

```
Commande confirmee : Frequence LoRa : 868.0 MHz
```

## 3. Documentation d'Utilisation

### 3.1 Démarrage du Système

Après téléversement du firmware, la procédure de démarrage se déroule automatiquement :

N°	Étape	Indication de succès
1	EEPROM — Chargement config	Configuration réseau chargée
2	SPI — Initialisation	Bus SPI démarré (D4 SD, D10 ETH)
3	Micro SD — Détection	Partition FAT32 montée
4	LCD — Initialisation	Écran allumé, affichage du boot
5	Ethernet — Initialisation	IP obtenue (DHCP ou statique)
6	Telnet — Serveur	Port 23 en écoute
7	FTP — Serveur	Port 21 en écoute (si SD présente)
8	Serveur WEB — Start	LED D5 allumée, port 80 actif
9	LoRa — Récepteur	868 MHz Serial1 D0/D1 initialisé
10	Open-Meteo — 1re connexion	Données météo internet chargées

#### LED D5 — Indicateur de statut

La LED D5 s'allume uniquement lorsque le serveur WEB est démarré avec une configuration réseau valide chargée depuis l'EEPROM. Si elle ne s'allume pas, vérifier le câble RJ45 et l'IP configurée.

### 3.2 Connexion au Site WEB

Le serveur WEB intégré permet de consulter les données météo en temps réel depuis n'importe quel navigateur du réseau local.

#### Accès au Site

Ouvrir un navigateur web et saisir l'adresse IP de l'Arduino dans la barre d'adresse :

**http://192.168.0.10**

(remplacer par l'IP configurée avec la commande ip)

#### Pages Disponibles

http://[IP]/ ou http://[IP]/index.htm : page principale météo (servie depuis la SD)

http://[IP]/ConfIP.htm : page de configuration IP réseau (interface graphique)

#### Authentification (si activée)

**Login par défaut :** user

**Mot de passe par défaut :** pass

**Format EEPROM :** dXNlCjpwYXNz (encodage Base64 de user:pass)

Pour modifier les identifiants, encoder login:motdepasse sur base64encode.org puis :

```
> login-passWEB YWRtaW46MDFfQWRtaW4=
> save
```

### 3.3 Interface Web — Page Principale

La page index.htm est une interface responsive au thème sombre (palette IoToS). Elle affiche deux blocs de données en temps réel :

#### Bloc 1 — Capteurs LoRa Locaux (rafraîchissement 1 seconde)

- Vent : vitesse en km/h
- Direction : girouette SVG animée — 8 directions (N, NE, E, SE, S, SO, O, NO)
- Pluie : pluviomètre en mm
- Température : en °C (bleu < 0°C, orange > 30°C)
- Humidité : en %RH
- Pression atmosphérique : en hPa
- RSSI LoRa : qualité du signal radio en dBm
- Badge statut : vert OK / rouge hors ligne

#### Bloc 2 — Open-Meteo Internet (rafraîchissement 30 secondes)

- Température, température ressentie, point de rosée
- Humidité, pression, vent (vitesse + direction + rafales)
- Précipitations (pluie totale, pluie liquide, averses, neige)
- Couverture nuageuse, visibilité, rayonnement solaire
- Code météo WMO (0=ciel clair, 61=pluie légère, 95=orage...)
- Horodatage de la dernière mise à jour Open-Meteo

### 3.4 Utilisation de l'Écran LCD

L'écran LCD 128x64 affiche les informations de façon autonome. Navigation avec les 3 boutons Grove.

Bouton	Port	Fonction
Bouton OK	D8	Valider la sélection du menu
Bouton (+)	D2	Défiler vers le bas dans le menu
Bouton (-)	D3	Défiler vers le haut dans le menu

Le menu LCD propose jusqu'à 5 choix par page sur plusieurs pages. Un cadre indique la sélection courante. Les 3 boutons utilisent les interruptions hardware pour une réponse instantanée.

## 3.5 Accès FTP à la Carte SD

**Logiciel recommandé** : FileZilla (gratuit)

**Hôte** : IP de l'Arduino (ex. 192.168.0.10) — Port : 21

**Login FTP** : arduino — Mot de passe FTP : arduino (par défaut)

### Configuration FileZilla

Fichier → Gestion de sites → Paramètres de Transfert → Limiter les connexions simultanées à 1. Obligatoire.

Commandes CLI de gestion de fichiers SD :

Commande	Syntaxe	Description
<code>dir / ls</code>	<code>dir</code>	Lister les fichiers de la carte SD
<code>type</code>	<code>type meteo.csv</code>	Afficher le contenu d'un fichier
<code>del</code>	<code>del ancien.log</code>	Supprimer un fichier
<code>rename</code>	<code>rename a.csv b.csv</code>	Renommer un fichier
<code>copy</code>	<code>copy a.csv b.csv</code>	Copier un fichier
<code>edit</code>	<code>edit config.txt</code>	Éditer un fichier texte en ligne

## 3.6 Variables Météo LoRa

Les variables météo sont mises à jour en temps réel dans la boucle principale `loop()` :

Variable C++	Type	Unité	Description
<code>rx_vitesseVent</code>	<code>uint8_t</code>	<b>km/h</b>	Vitesse du vent 0–255
<code>rx_positionGirouette</code>	<code>uint8_t</code>	<b>1–8</b>	Direction : 1=N 2=NE ... 8=NO
<code>rx_pluviometre</code>	<code>uint8_t</code>	<b>mm</b>	Cumul de pluie 0–255
<code>rx_temperature</code>	<code>int8_t</code>	<b>°C</b>	Température -128 à +127
<code>rx_humidite</code>	<code>uint8_t</code>	<b>%RH</b>	Humidité relative 0–100
<code>rx_pression_hPa</code>	<code>uint16_t</code>	<b>hPa</b>	Pression atmosphérique
<code>rx_rssi</code>	<code>int16_t</code>	<b>dBm</b>	Signal LoRa reçu
<code>rx_trame_valide</code>	<code>bool</code>	—	true = trame CRC valide

## 3.7 Dépannage Rapide

Symptôme	Solution
<b>LED D5 éteinte après boot</b>	Vérifier le câble RJ45, l'IP et exécuter reset après save

Symptôme	Solution
<b>Site WEB inaccessible</b>	Vérifier l'IP avec print, tester ping depuis le CLI
<b>LoRa : aucune donnée reçue</b>	Vérifier câblage D0/D1, fréquence LoRa (LoRaFreq 868.0)
<b>SD non détectée</b>	Reformater en FAT32, réinsérer avant la mise sous tension
<b>FTP : erreur de transfert</b>	FileZilla : limiter à 1 connexion simultanée
<b>LCD : écran blanc</b>	Vérifier adresse I2C 0x3F, connecteur A4/A5
<b>Doubles caractères Telnet</b>	Taper cli off dans le terminal (MobaXterm)
<b>Open-Meteo sans données</b>	Vérifier la connexion Internet, attendre 5 min (délai API)