Interreg CENTRAL EUROPE

Co-funded by the European Union

DARKERSKY4CE

Photometer Network

Manual and SQM installation guide

Version 1

06 2025

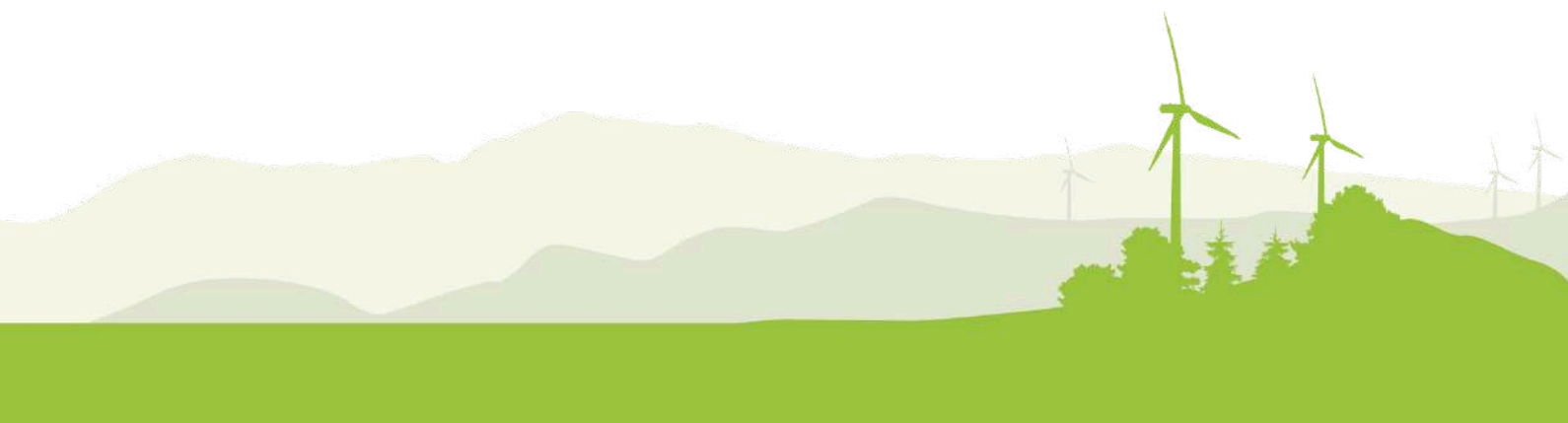


Table of Contents

Project Title	DARKERSKY4CE - Strategic Transnational Approach to Reduce Light pollution in Central Europe
Project ID and Acronym	CE0200765 - DARKERSKY4CE
Name of Project Partner Organisation	Istituto Nazionale di Astrofisica
Partner Number	LP

Purpose of this document

This document, titled *Photometer Network - Manual and SQM Installation Guide*, provides a comprehensive overview of the Photometer Network (D.1.3.4) developed within the DARKERSKY4CE project of the Interreg Central Europe programme. In addition to harmonising sky brightness and light pollution measurements through the DARKERSKY4CE Light Pollution Repository, the project also aims to enhance the spatial coverage and quality of these measurements across Central Europe by deploying new photometers.

The document begins with an introduction to the Sky Quality Meter (SQM) instruments and their technical specifications, followed by a description of the interface used to access and visualise data within the Light Pollution Repository. Finally, it presents a detailed, step-by-step installation guide for setting up an SQM-LE instrument to operate it autonomously and synchronise its data with the DARKERSKY4CE cloud storage.

This guide is designed to enable project partners, as well as anyone interested in contributing to this experimental effort, to replicate the installation process within a few hours, without requiring advanced technical skills.

Acknowledgments

This document is part of the INTERREG Central Europe project “DARKERSKY4CE - Strategic Transnational Approach to Reduce Light pollution in Central Europe”. The project is supported by the Interreg CENTRAL EUROPE Programme 2021-2027 with co-financing from the European Regional Development Fund (ERDF). The project total budget is 1,791,013.00 EUR.

1. Introduction

The Photometer Network activity of the DARKERSKY4CE project consists not only in the systematic organisation and harmonisation of sky brightness and light pollution measurements carried out in the region of Central Europe, but also in the **deployment of new photometers**.

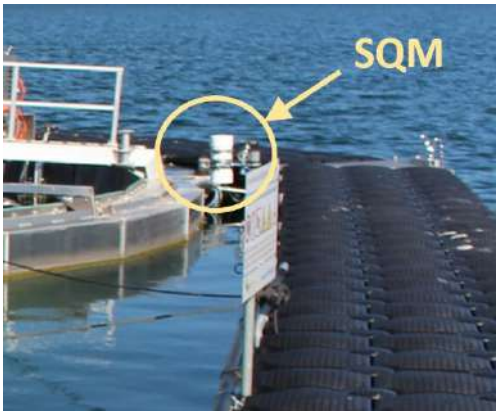
The DARKERSKY4CE Photometer Network is dedicated to improving the density and quality of sky brightness measurements and to **quantifying light pollution in non-urban areas**, with particular attention to the areas of the project Pilot Demo Sites.

1.1 The SQM Instrument

To this purpose, the DARKERSKY4CE partnership selected the **SQM** (**Sky Quality Meter**) device developed by Unihedron [Unihedron].

The Sky Quality Meter (SQM) is an affordable, portable, and easy-to-use instrument designed to measure night sky brightness in units of **magnitudes per square arcsecond** (also referred to as **MSAS**). SQMs are widely used by researchers and astronomers and are available in several variants. The model selected for the DARKERSKY4CE project is the **SQM-LE** [SQM-LE] (Ethernet-enabled), which allows for an **easy installation in a fixed position**. In this configuration, the SQM device is typically installed on the roof of a building (but other configurations are also possible, see Figure 1.1.1) by means of a dedicated support, enclosed in a waterproof housing, and connected via an Ethernet cable running to a mini-PC, which is instead installed in an indoor room to be kept safe from the elements.

Figure 1.1.1 - SQM kit components and installation configuration. Left: a picture of the SQM kit provided by Unihedron [SQM-LE]. Right: Sky quality meter (SQM) mounted in a weatherproof housing on the floating platform at the experimental facility (LakeLab) in Lake Stechlin, Germany [Jechow et al. 2016].



SQMs are well-characterised instruments, thanks to the dedicated calibration performed by Unihedron before shipping each unit, as well as thanks to the extensive studies conducted at the Light Pollution Photometry and Radiometry Laboratory (LPLAB) of the Light Pollution Science and Technology Institute [Cinzano 2005].

In terms of technical specifications, the Field of View (FoV) of the SQM-LE is approximately 20°, defined by the Full Width at Half Maximum (FWHM) of the relative radiance as a function of the incident angle. SQMs measure visible light across the 350 nm to 700 nm wavelength range, spanning from near-ultraviolet to red, with peak sensitivity around 500 nm (green).

1.2 The DARKERSKY4CE Photometer Network

The DARKERSKY4CE project plans the installation of **at least four SQM instruments**: one in Italy (initially used for testing purposes), two in Hungary, and one in Germany. Additional instruments may be deployed by other project partners over the course of the project. An **updated map of the instruments** installed within the Photometer Network is available on the dedicated webpage [Photometer Network] of the **DARKERSKY4CE Light Pollution Repository** [LP Repository], hosted and maintained by INAF (see Figure 1.2.1).

Figure 1.2.1 - Screenshot of the Photometer Network webpage on the DARKERSKY4CE Light Pollution Repository (darkesky4ce.inaf.it), with the links to browse and download the raw data acquired by the SQM instruments.

DATA

Photometer Network

The Photometer Network activity of the DARKERSKY4CE project consisted not only in the systematic organisation and harmonisation of sky brightness and light pollution measurements carried out in the region of Central Europe, but also in the deployment of new photometers.

The DARKERSKY4CE Photometer Network is dedicated to improving the density and quality of sky brightness measurements and to quantifying light pollution in non-urban areas, with particular attention to the areas of the project Pilot Demo Sites.

To this purpose, the DARKERSKY4CE partnership selected the SQM (Sky Quality Meter) device developed by Unihedron. This is an affordable instrument, widely used among researchers and astronomers, to measure night sky brightness in units of magnitudes per square arcsecond. Indeed, SQMs measure the visible light in the wavelength range from 300 to 700 nm, integrating incoming light in a field of view of about 20°.

The Unihedron SQM-LE (Ethernet-enabled) unit allows for an easy installation in a fixed position. In this configuration, the SQM device is typically installed on the roof of a building by means of a dedicated support, enclosed in a waterproof housing, and connected via an Ethernet cable running to a mini-PC, which instead is installed in an indoor room to be kept safe from the elements. The DARKERSKY4CE SQMs are operated via a version of the open-source software named PySQM that was adapted to the needs of the project itself.

BROWSE THE DATA OF THE NETWORK

HERE

DOWNLOAD THE RAW DATA

HERE

The DARKERSKY4CE SQMs are operated via a version of the open-source software named **PySQM** [PySQM Alteholz] that was adapted to the needs of the project itself [PySQM INAF]. This software, written in Python, manages **data acquisition at regular intervals** and allows scheduling of observations based on the Sun's elevation at the installation site. Data acquisition is performed in the format of the **community standard for skyglow observations** [DarkSky 2013]. PySQM also automatically generates daily plots of MSAS values and performs preliminary statistical analyses of nightly data.

Data from each instrument of the DARKERSKY4CE Photometer Network are continuously synchronised to a **shared, openly accessible cloud storage**. From the Photometer Network webpage, these data are **available for download** in TXT format (daily and monthly raw data) and PNG format (daily plots). Additionally, an **interactive web tool** is provided to browse, visualise, and interpret daily MSAS data along with the related metadata, such as Sun and Moon elevation plots (see Figure 1.2.2).

Figure 1.2.2 - Interactive web interface made to browse, visualise, and interpret daily data of sky brightness acquired by SQMs of the DARKERSKY4CE Photometer Network, along with related metadata. The data visualised here are from the SQM installed indoors at the INAF - Astrophysical Observatory of Torino for testing purposes.

2. SQM Installation Guide

2.1 Hardware and Software Requirements

a) Mini-PC with Microsoft Windows OS and its power adapter;
b) SQM kit, including:
   i) SQM-LE instrument
   ii) installation USB key
   iii) power adapter
   iv) outdoor-rated Cat6 Ethernet cable
   v) waterproof housing
   vi) Power over Ethernet (PoE) kit
c) access to three power outlets and a wired or wireless Internet connection;
d) USB-A mouse and keyboard;
e) HDMI screen (or an alternative display connection, provided you have the appropriate adapter for the mini-PC).

2.2 Setup Guide

2.2.1 Hardware Setup

Connect the 🔌`mini-PC to power` using the provided adapter, then connect a 🖱️`mouse and keyboard` to the mini-PC via two USB-A ports, and a 🖥️`monitor` to the HDMI port. These accessories are needed only during the configuration phase; the system can operate without them under normal working conditions.

2.2.2 Mini-PC Installation

Power on and initiate the 🛠️`initial installation` of the mini-PC with Windows OS, preferably version 11. The Windows Office program suite is not necessary for system operation; therefore, you can skip the free trial during setup.

During the installation phase (and during standard operating conditions), you will need to 🛜`connect to an Internet source`, either wireless or wired. You can choose whichever connection method you prefer at this stage. If you choose a wireless connection, be sure that the option ✅`Connect automatically` is selected. For the final system configuration, one Ethernet port will be connected to the SQM. If your mini-PC has two or more Ethernet ports, use a wired connection; otherwise, a Wi-Fi connection will suffice.

Set up a local account or one linked to a Microsoft account to serve as the system administrator.

2.2.3 System Optimisations

Perform the following system configurations:

a) Set the **system language to English (UK)**. This will help you follow this installation guide. To do so, follow these steps:
   i) Open the Windows ➡️`Settings` and go to ➡️`Time & language > Language & region`.
   ii) Set 🔍`Windows display language` to 👆`English (United Kingdom)`.
b) Set the **timezone to UTC**. To do so, follow these steps:
   i) Open the Windows ➡️`Settings` and go to ➡️`Time & language > Date & time`.
   ii) Under 🔍`Time zone`, select 👆`(UTC) Coordinated Universal Time`.
c) **Disable sleep mode**. To do so, follow these steps:
   i) Open the ➡️`Control Panel` and search for 🔍`Power Options` in the search bar (top right).
   ii) Under ➡️`Choose or customize a power plan > Preferred plans`, click 👆`Change plan settings` for the selected plan.
   iii) Set both 🔍`Turn off the display` and 🔍`Put the computer to sleep` to 👆`Never`.
d) **Show hidden files and file extensions** in File Explorer. To do so, follow these steps:
   i) Open a window of the 📂`File Explorer`.
   ii) In the top settings bar, 👆`View > Show`.
   iii) Tick both ✅`File name extensions` and ✅`Hidden items`.

2.2.4 Automatic Restore of Mini-PC on Power Loss

To set up the mini-PC to automatically turn back on after a power outage, start by ⚠️`rebooting the mini-PC`. As soon as the manufacturer's logo appears on the screen, repeatedly press the ⬇️`Esc` key within a few seconds until the **BIOS menu** appears. Once in the BIOS, navigate to the ➡️`Chipset` tab, locate the 🔍`PCH-IO` option, and set it to ✅`S0 State`. After making this change, ✅`save` the settings and ⬅️`exit` the BIOS. The mini-PC will then reboot automatically.

To verify that the settings were successfully applied, unplug the mini-PC while it is powered on. After a few seconds, plug the power adapter back in. The mini-PC should **power on automatically without requiring you to press any buttons**.

Please note that the **instructions provided above may vary depending on the manufacturer** of your mini-PC. The steps provided here refer specifically to the model recommended for the installation of the SQM kits in the first stations of the DARKERSKY4CE Photometer Network (manufacturer: GMKtec), but all machines will have

an equivalent option to set up the automatic restore on power loss. Visit the manufacturer's website to learn more.

2.2.5 Download and Enable Autologon

The programs running the SQM in the background cannot operate if the user is not logged in. Therefore, it is necessary to configure the mini-PC to automatically log in after every boot.

To enable this, download the 🌐Autologon Tool from the Windows Learn webpage. Once downloaded, unzip the file into a folder (for example ✍Autologon) and move this folder to 📁C:\Program Files. Moving the folder will require **administrator privileges**. Then, run 💡Autologon.exe by double-clicking it and follow the on-screen instructions. At the end of the process, you will be asked to enter the Windows user credentials, which will be securely encrypted and stored to enable automatic logon.

It is also recommended to **disable the Windows Hello feature**. To do this, open the Windows ➡Settings and go to ➡Accounts > Sign-in options and disable both ❌Facial recognition (Windows Hello) and ❌Fingerprint recognition (Windows Hello). You may choose to keep the ✅PIN (Windows Hello) option if preferred.

To verify that the Autologon tool is working correctly, reboot the mini-PC. When the Windows login screen appears, you should see an automatic login attempt for your user account. If the login fails, double-check that you entered the correct username and password when configuring the Autologon tool.

2.2.6 Activate Automatic Lock Session after First Logon at Boot

To **safeguard the integrity of the system** after the automatic login at boot, it is recommended to configure an automatic session lock using the Windows Task Scheduler.

In order to configure this task, search for 🔍Task Scheduler in the Windows search bar and open it. In the 🗒Actions panel on the right, select 👆Create Task. A new window will appear where you will need to fill in the following information:

a) Under the ➡General tab, configure the following entries:
   i) 🔍Name: type ✍AutoLockAfterBoot.
   ii) 🔍Description: type ✍Automatic lock of the user session 30 seconds after autologon at boot.
   iii) Under the 🗒Security Options section, tick ✅Run only when the user is logged on and ✅Run with the highest privileges.
   iv) 🔍Configure for: select 👆Windows 10 from the dropdown menu.
b) Under the ➡Trigger tab, click 👆New to create a new trigger. In the window that appears, configure the following settings:

i) 🔍Begin the task: select 👆At logon, then select 👆Specific User and choose the correct user account (you will typically have only one user; if not, ensure you select the appropriate one).
ii) Tick the option ✅Delay task for and select 👆30 seconds from the dropdown menu.
iii) Click 👆Ok to save the trigger.

c) Under the ➡️Actions tab, click 👆New to create a new action. In the window that appears, configure the following settings:
i) 🔍Action: select 👆Start a program.
ii) 🔍Program/script: type ✍️rundll32.exe.
iii) 🔍Add arguments (optional): type ✍️user32.dll,LockWorkStation.
iv) Click 👆Ok to save the action.

d) Under the ➡️Conditions tab, uncheck both ❌Start the task only if the computer is on AC power and ❌Stop if the computer switches to battery power options. This ensures the task will run regardless of the power source.

e) Click Ok to save the task.

To verify that the task is operating correctly, reboot the mini-PC. After the system automatically logs in, wait for 30 seconds. The user session should automatically lock, and you should then be prompted to enter the password or PIN to regain access.

2.2.7 Software Installation

Install the following programs to support the configuration and operation of the system:

a) **Web browser**. The default browser (Microsoft Edge) is acceptable, but it is recommended to install 🌐Google Chrome for better compatibility with some tools required in the final system configuration, such as Chrome Remote Desktop.

b) **Windows Terminal**. This program should be pre-installed in all Windows 11 versions. If it is not available, install it from the Microsoft Store. It will be used to execute **command-line operations** during the installation process.

c) 🌐Miniconda, a miniature Anaconda distribution for Python. Download and launch the installer, following the provided instructions. Miniconda and Python will be used to **run the PySQM program** to operate the SQM. Take note of the following important details during the Miniconda installation:
i) When prompted to select an installation location, choose ✍️C:\ProgramData. This ensures that the Python environment will be accessible to all users, if multiple accounts are present;
ii) Near the end of the installation process, make sure to select the options ✅Register Miniconda as the system Python X.X and ✅Clear the package cache upon completion.

d) 🌐Notepad++, an advanced text and code editor. Download and launch the installer, and follow the provided instructions using the default settings. This editor will be used to **modify the PySQM code and settings**. Alternatively, you can use Visual Studio Code if preferred.
e) 🌐NSSM, the Non-Sucking Service Manager. It will be downloaded as a ZIP archive. 🤏Extract the contents and move the resulting folder to 📁C:\Program Files. Moving the folder will require administrator privileges. NSSM will be used to **configure the background services** needed to run PySQM and continuously upload the SQM data online.
f) 🌐Rclone, a command-line program to **manage files on cloud storage**. From the latest release, choose the 👆Intel/AMD - 64 Bit version. It will be downloaded as a ZIP archive. 🤏Extract the contents and move the resulting folder to 📁C:\Program Files. Moving the folder will require administrator privileges. Rclone will be used to **synchronise the SQM data** with a public cloud storage service.

2.2.8 Environment Variables Configuration

To proceed, we need to ensure that the operating system can **locate the installation files for Miniconda, NSSM, and Rclone**.

To do this, follow these steps:

a) Open the Windows ➡️Control Panel and search for 🔍Environment Variables using the search bar in the top right corner.
b) Under the ➡️System section, click on 👆Edit the system environment variables. Administrator rights will be required to proceed. A new window will appear.
c) In this window, go to the ➡️Advanced tab and click the 👆Environment Variables button. Under the ➡️System variables section, find and select the 👆Path entry, then click the 👆Edit button.
d) A new window will appear. You will need to ➕add five new entries to the table displayed in this window. To do this, click 👆New and type each path one at a time. The paths to be added are as follows[1]:
   i) ✍C:\ProgramData\miniconda3
   ii) ✍C:\ProgramData\miniconda3\Scripts
   iii) ✍C:\ProgramData\miniconda3\Library\bin
   iv) ✍C:\Program Files\rclone-v1.69.2-windows-amd64
   v) ✍C:\Program Files\nssm-2.24\win64
e) To save the modifications, close all the windows by clicking 👆Ok at each step.

[1] Your actual paths may vary slightly depending on your specific installation setup.

2.2.9 SQM Installation

Connect the 🔌`SQM to power` using the provided power adapter. The SQM will power on automatically. Next, connect the 🔌`SQM to the mini-PC` using the supplied Ethernet cable.

In the final system configuration, you may prefer to use the 🌐`Power over Ethernet (PoE)` **kit** to connect the SQM to both the mini-PC and the power outlet. If you plan to use this setup, it is recommended to install the system in this configuration from the beginning, including the **outdoor-rated Cat6 Ethernet cable**.

To install the SQM outdoor in its final configuration, you need to install it with the provided weatherproof housing. Detailed instructions are available 🌐here.

Important: when connecting the Ethernet cable to the mini-PC, always remember to use the 🟥`Ethernet inverter` (the red adapter included in the SQM kit).

2.2.10 UDM Installation

UDM (**Unihedron Device Manager**) is the default application provided by Unihedron for managing the SQM and acquiring data from it.

In order to install UDM, insert the ➡️`Unihedron USB key` provided with the SQM kit into a USB-A port on the mini-PC. Open the 📁`File Explorer` and navigate to the USB key location. It will appear as a removable drive, typically labeled something like 📁`E:\`.

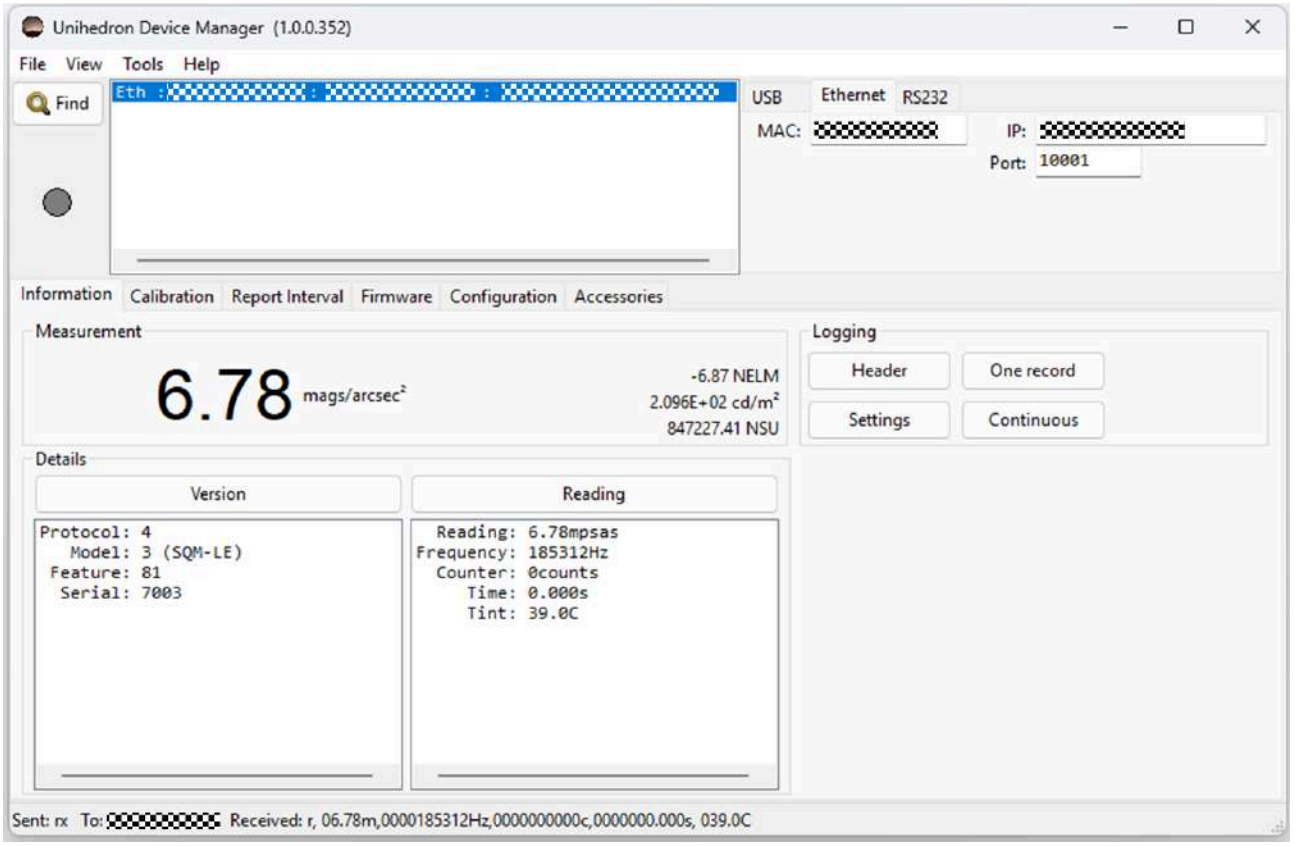
Inside the USB key, open the 📁`Windows` folder and run the installer named 💡`setup1.0.0.352.exe` by double-clicking it. Follow the installer steps, keeping all the predefined options.

Next, search for the 🔍`UDM app` (for example, using the Windows search bar) and launch it. If the SQM is correctly connected to the mini-PC, you should see a window similar to the one reported in Figure 2.2.1.

If the SQM is not detected, verify the Ethernet and power connections. You may also try rebooting both the SQM (by unplugging and plugging it back in) and the mini-PC.

Once the SQM is properly recognised, take note of its 🌐`IP address` (in the format of XXX.YYY.ZZZ.AAA) as displayed in the UDM app.

Figure 2.2.1 - Screenshot of the Unihedron Device Manager utility to operate the SQM.



2.2.11 SQM Fixed IP Configuration and Timeout Settings

Open a 🌐`web browser` and enter the ✍`SQM IP address` into the address bar at the top of the window, then press ⬇️`Enter`. A login pop-up will appear asking for a username and password. Leave both fields blank and click 👆`Sign in`.

You will access the **XPort Lantronix browser-based configuration interface**. From the side menu, select 👆`Network`, and under the ➡️`IP Configuration` section, choose ✅`Use the following IP Configuration`. Then enter the following details:

a) IP Address: ✍`169.254.65.200`
b) Subnet Mask: ✍`255.255.0.0`
c) Default Gateway: ✍`0.0.0.0`
d) DNS Server: ✍`0.0.0.0`

This configuration can be the same for every SQM, as it applies only to the local network between the mini-PC and the SQM instrument.

To save the changes, click 👆`Ok`, then click 👆`Apply Settings` in the side menu. A new page will appear. Wait for the process to complete, then close the browser.

From this point forward, you should access the SQM browser-based configuration interface by **entering this IP address in your web browser** address bar. To confirm the configuration was successful, launch the UDM app again and verify that the SQM IP address is updated to 169.254.65.200.

Reaccess the browser-based configuration interface. This time, from the side menu, navigate to ➡️`Channel 1 > Connection`. Under the ➡️`Disconnect Mode` section, set the 🔍`Inactivity Timeout` to ✍️`10 : 00` (mins : secs).

To save the changes, click 👆`Ok`, then click 👆`Apply Settings` in the side menu. A new page will appear. Wait for the process to complete, then close the browser.

2.2.12 PySQM Installation and Configuration

First, create a folder named 📁`SQM` in the root directory (typically 📁`C:\`). Inside the SQM folder, create another folder named 📁`data` , resulting in 📁`C:\SQM\data`.

Then, download the 🌐`PySQM software` from the INAF GitLab repository. To do this, click the 👆`Code` button and select 👆`Download source code > zip`. Then, 👊`unzip` the downloaded file, rename the extracted folder to ✍️`pysqm` and move it to 📁`C:\SQM`.

Next, you need to install the **required Python package dependencies** to run PySQM. To do this, open ➡️`Windows Terminal (PowerShell) as an Administrator` (you can search for it in the Windows search bar, then right-click and select 👆`Run as administrator`).

In the terminal, type the following commands[2] one at a time, pressing ⬇️`Enter` after each:

a) ✍️`conda install numpy`
b) ✍️`conda install ephem`
c) ✍️`conda install matplotlib`
d) ✍️`conda install pip`

For each package, wait for the process to complete and, when prompted, confirm the installation by typing ✍️`y` or ✍️`yes` in the terminal.

Finally, you need to **edit the PySQM configuration** reported in the 📝`config.py` file inside the PySQM folder. Open the file with Notepad++ and carefully edit the following entries, using the provided examples as a guide. Make sure to ✅`save` the file after editing.

a) ➡️`SITE location` section. Update the site-specific information:

[2] If you encounter an error while attempting to install these Python packages through conda, try executing the commands *conda init powershell* and *conda config --set auto_activate_base false*. Then, restart Windows Terminal (PowerShell) and try the package installation again.

i) ✍_observatory_name = 'IT_PinoTorinese'
The first two letters indicate the country (ISO code), followed by an underscore and the site/observatory name, usually the municipality (without spaces).

ii) ✍_observatory_latitude = 45.039868
Latitude North of the site in decimal degrees (use at least five decimal digits).

iii) ✍_observatory_longitude = 7.762582
Longitude East of the site in decimal degrees (use at least five decimal digits).

iv) ✍_observatory_altitude = 600
Altitude of the site in meters above sea level.

b) ➡DEVICE specs section. Provide detailed information about the device and the data contact person:

i) ✍_device_locationname = 'Pino Torinese/Italy - Astrophysical Observatory of Torino'
Detailed location description.

ii) ✍_data_supplier = 'Dario Barghini / INAF - OATo'
Contact details of the data supplier or person responsible for the instrument.

iii) ✍_device_addr = '169.254.65.200'
Local IP address assigned to the SQM instrument.

c) ➡MEASURE settings section. Set the measurement parameters:

i) ✍_local_timezone = +1
Local timezone of the site, ignoring daylight saving time (e.g. Central Europe is always +1).

ii) ✍_offset_calibration = -0.11
Calibration offset (in mag/arcsec^2) to account for the SQM housing window, as reported in the datasheet provided with the SQM kit.

Leave all other parameters untouched. To **test the correct functionality of PySQM**, open ➡Windows Terminal (PowerShell). Navigate to the PySQM folder by typing the following command:

✍cd C:\SQM\pysqm

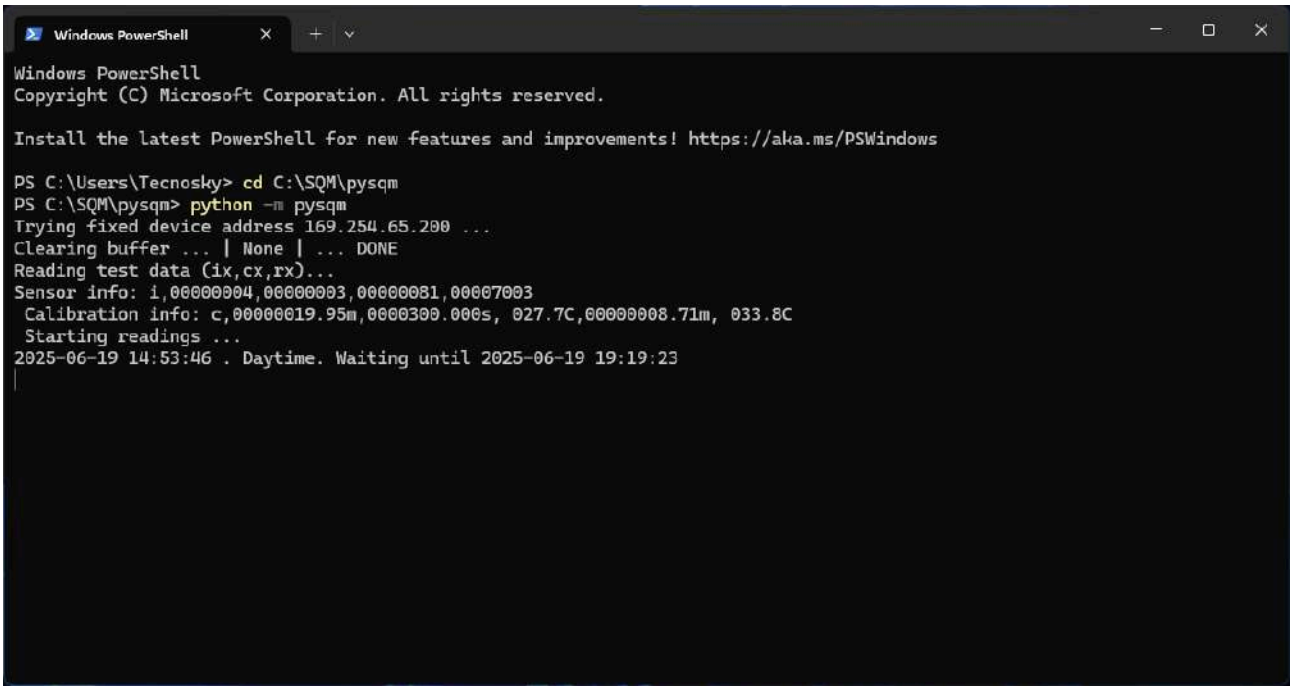
and press ⬇Enter. Then, start the program by typing:

✍python -m pysqm

and press ⬇Enter. The PySQM program will launch, and you should see a console log similar to the example shown in Figure 2.2.2. During the daytime, the program will

remain idle and wait for nighttime. Once nighttime begins, you should see measurements being automatically recorded every 5 minutes.

Figure 2.2.2 - Screenshot of Windows Terminal (PowerShell) showing PySQM running during daytime.



After verifying that PySQM is running correctly, stop its execution by simultaneously pressing the ⬇️`Ctrl + C` keys on the keyboard.

To run PySQM from any directory without having to navigate to its folder each time, you need to **install PySQM as a Python package**. To do this, open a ➡️`Windows Terminal (PowerShell) as an Administrator` and navigate again to the 📁`PySQM folder`, then type the following command:

✍️`pip install -e .`

Press ⬇️`Enter` and wait for the installation to complete. You can still modify the 📝`config.py` file and/or update the code even after installing the package, since PySQM was installed using the -e (editable/development) option. Any changes made to the configuration file will be immediately applied the next time you run PySQM.

2.2.13 RClone Configuration

One of the final steps in the system configuration is to set up Rclone to **connect to the** ☁️`DARKERSKY4CE Photometer Network cloud storage`. Before starting this process, ensure that you have the necessary credentials for the account with access to said cloud storage.

Start by opening a ➡️`Windows Terminal (PowerShell) as an Administrator` and type the following command:

✍`rclone config`

Press ⬇`Enter` to start the Rclone configuration tool. When prompted, type ✍`n` and press ⬇`Enter` to create a new remote. You will then be asked to provide the following information. Type each response as shown below, pressing Enter after each entry:

a) *name*> ✍`gdrive`
b) *Storage*> ✍`20` (this corresponds to Google Drive)
c) *client_id*> ⬇`Enter`
d) *client_secret*> ⬇`Enter`
e) *scope*> ✍`1` (this gives full access to the drive)
f) *service_account_file*> ⬇`Enter`
g) *Edit Advanced config? y/n*> ✍`n`
h) *Use web browser to automatically authenticate rclone with remote? y/n*> ✍`y`
   At this point, a web browser window will open. 👤`Authenticate` using the account with access to the shared DARKERSKY4CE Photometer Network.
i) *Configure this as a Shared Drive*> ✍`y`
   Select 👆`DARKERSKY4CE Photometer Network` from the list of available shared drives.
j) *Keep the "gdrive" remote*> ✍`y`

You should now see the new remote listed under the ➡`Current remotes` section in the terminal. To exit the Rclone configuration tool, type ✍`q` and press ⬇`Enter`.

2.2.14 NSSM Service Configuration

The final step in configuring the SQM node is to **set up the two background services**: one to operate the SQM via PySQM, and the other to synchronise the data to the shared DARKERSKY4CE cloud storage.

Let's begin by installing the first service. First, create a folder named 📁`logs` inside the 📁`C:\SQM\` directory. Then, open a ➡`Windows Terminal (PowerShell)` and type the following command:

✍`nssm install pysqm`

Press ⬇`Enter` to launch the NSSM service configuration tool. Administrator rights will be required to proceed. A small configuration window will appear, containing several tabs that need to be filled out. Enter the following parameters in the respective tabs:

a) Under the ➡`Application` tab, configure the following entries:
   i) 🔍`Path`: type ✍`C:\ProgramData\miniconda3\python.exe`.
   ii) 🔍`Startup directory`: type ✍`C:\ProgramData\miniconda3` (it should be automatically filled in by NSSM).
   iii) 🔍`Arguments`: type ✍`-m pysqm`.

b) Under the ➡️Details tab, configure the following entries:
   i) 🔍Display name: type ✍️PySQM.
   ii) 🔍Description: type ✍️PySQM service to operate the SQM-LE of the DARKERSKY4CE Photometer Network.
c) Under the ➡️I/O tab, configure the following entries:
   i) 🔍Output (stdout): type ✍️C:\SQM\logs\pysqm_stdout.log.
   ii) 🔍Error (stderr): type ✍️C:\SQM\logs\pysqm_stderr.log.
d) Under the ➡️File rotation tab, configure the following entries:
   i) Tick the options ✅Rotate files and ✅Rotate while service is running.
   ii) 🔍Restrict rotation to file bigger than: ✍️1000000 bytes.

Leave all other parameters unchanged. Click on the 👆Install service button to create the PySQM service.

In order to verify that the service was correctly installed, open the ➡️Task Manager (for example, by searching for it in the Windows search bar) and select the ➡️Services tab from the sidebar. You can quickly find the service by typing ✍️PySQM in the search bar at the top of the window. The service status should initially appear as 🔍Stopped.

To start the service, right-click on the 👆PySQM entry in the list and select 👆Start. After a few seconds, the service status should update to 🔍Starting, and then to 🔍Running. From now on, the service will **automatically start** each time the mini-PC boots (for example, after a power outage).

To install the Rclone NSSM service, the first step is to create a script that will handle the **synchronisation between the local and remote folders** using Rclone. To do this, open ➡️Notepad++ (for example, by searching it on the Windows search bar). A new file will be automatically created. Type the following code into the editor:

```
✍️@echo off
✍️:start
✍️rclone copy “C:\SQM\data” “gdrive:Data/SQM_[_observatory_name]”
--fast-list --config “C:\Program
Files\rclone-v1.69.2-windows-amd64\rclone.conf”
✍️timeout /t 300 >nul
✍️goto start
```

Instead of ✍️[_*observatory_name*], you need to insert the name you defined in the 📝config.py file (see Section 2.2.12). In the example used here, it would be: ✍️IT_PinoTorinese.

Most importantly, you must ensure that the corresponding folder, in this case 📁SQM_IT_PinoTorinese, is present in the cloud storage within 📁SQM_Data.

Then, save the file as a batch script with the name 📝`rclone.bat` in the folder 📁`C:\Program Files\rclone-v1.69.2-windows-amd64.` Also, make sure that the file 📝`rclone.conf` is correctly located in the same folder.

Then, follow the same steps performed to install the PySQM service, this time typing:

✍`nssm install rclone`

and entering the following parameters in the respective tabs:

e) Under the ➡`Application` tab, configure the following entries:
   i) 🔍`Path`: type ✍`C:\Program Files\rclone-v1.69.2-windows-amd64\rclone.bat`.
   ii) 🔍`Startup directory`: type ✍`C:\Program Files\rclone-v1.69.2-windows-amd64` (it should be automatically filled in by NSSM).
   iii) 🔍`Arguments`: type ✍`-m pysqm`.
f) Under the ➡`Details` tab, configure the following entries:
   i) 🔍`Display name`: type ✍`RClone`.
   ii) 🔍`Description`: type ✍`Automatic RClone copy of SQM data to DARKERSKY4CE Photometer Network cloud storage`.
g) Under the ➡`I/O` tab, configure the following entries:
   i) 🔍`Output` (stdout): type ✍`C:\SQM\logs\rclone_stdout.log`.
   ii) 🔍`Error` (stderr): type ✍`C:\SQM\logs\rclone_stderr.log`.
h) Under the ➡`File rotation` tab, configure the following entries:
   i) Tick the options ✅`Rotate files` and ✅`Rotate while service is running`.
   ii) 🔍`Restrict rotation to file bigger than`: ✍`1000000` bytes.

Leave all other parameters unchanged. Click on the 👆`Install service` button to create the RClone service.

Finally, follow the same instructions described above for the PySQM service to verify the correct installation of the RClone service and to start it.

2.2.15 Configuring Remote Access to the SQM Node

The final step is to configure remote access to the mini-PC connected to the SQM. A simple and effective solution is to use 🌐Chrome Remote Desktop. The installation process is straightforward and fully guided. Alternatively, you may use other remote access providers if preferred.

For this purpose, you can use the same account that was used to configure the Rclone service. It is recommended to set up **PIN access to secure the remote connection** through this channel.

2.3 Final Steps and Maintenance of the SQM Node

Once the installation process is complete, you will need to keep the mini-PC powered on to continue acquiring data and syncing it to the cloud storage. In order to **maintain system security** while allowing these processes to run, you can configure a **shortcut to lock the user session**.

In order to do this, right-click on the 🖥️`Desktop` and select 👆`New > Shortcut`. A new window will appear. Under 🔍`Type the location of the item`, enter the following command:

✍️`rundll32.exe user32.dll,LockWorkStation`

Then, click ⬇️`Next`. Under 🔍`Type a name for this shortcut`, enter ✍️`Lock Session`, and press ⬇️`Finish`.

A new shortcut will now appear on your desktop. By double-clicking this shortcut, the current user session will be instantly locked, and you will be required to enter your password or PIN to regain access. In the meantime, all background processes will continue to run without interruption.

If you wish, you can customise the shortcut icon by right-clicking on the shortcut and selecting 👆`Properties`. Then, click the 👆`Change Icon` button. You can either choose a lock icon from the suggested list or browse your system to select a different icon. Confirm the change by clicking 👆`Ok` twice.

In order to verify that the entire process was successful, access the ☁️`DARKERSKY4CE Photometer Network cloud storage` and check whether data from your station are being uploaded. Please note that data acquisition and upload will only occur during the night. This check should be performed **periodically to ensure that no disruptions** to the normal operation of the system have occurred.

You can also verify the status of the services by checking the **NSSM log files** located in the folder 📁 `C:\SQM\logs`. These log files can be easily opened with Notepad++.

It may occasionally be necessary to **update the PySQM software** to a newer version. To do this, first open the ➡️`Task Manager` and select the ➡️`Services` tab from the sidebar. Locate the 👆`PySQM` service, right-click on it, and select 👆`Stop`.

Once the service is stopped, you can proceed with updating the code, for instance by replacing the folder 📁`C:\SQM\pysqm` with the new version.

After the update, return to the Services tab of the Task Manager, search for the 👆`PySQM` service again, right-click on it, and select 👆`Start` to restart the service.

3. References

3.1 Bibliography

Cinzano 2005 - Cinzano P., *Night Sky Photometry with Sky Quality Meter*, ISTIL Internal Report n. 9, v.1.4 2005
https://unihedron.com/projects/darksky/sqmreport_v1p4.pdf

DarkSky 2013 - DarkSky International, *Definition of the community standard for skyglow observations*, Version 1.0, (10 April 2013)
https://darksky.org/app/uploads/bsk-pdf-manager/47_SKYGLOW_DEFINITIONS.PDF

Jechow et al. 2016 - Jechow A., et al., *Evaluating the summer night sky brightness at a research field site on Lake Stechlin in northeastern Germany*, Journal of Quantitative Spectroscopy and Radiative Transfer, Vol. 181, pp. 24-32, (2016)
http://dx.doi.org/10.1016/j.jqsrt.2016.02.005

3.2 Webliography

Autologon Tool - Windows tool to enable automatic logon at boot
https://learn.microsoft.com/it-it/sysinternals/downloads/autologon

Chrome Remote Desktop - Main webpage of the Chrome Remote Desktop tool
https://remotedesktop.google.com

Google Chrome - Download page of the Google Chrome browser
https://www.google.it/intl/en/chrome/

LP Repository - DARKERSKY4CE Light Pollution Repository main webpage
https://darkersky4ce.inaf.it/

Miniconda - Download page of the Miniconda Python distribution
https://www.anaconda.com/download

Notepad++ - Download page of the Notepad++ text editor
https://notepad-plus-plus.org/downloads/

NSSM - Download page of the Non-Sucking Service Manager for Windows OS
https://nssm.cc/download

Photometer Network - DARKERSKY4CE Photometer Network webpage
https://darkersky4ce.inaf.it/photometer-network/

PySQM Alteholz - Python3 client for the Sky Quality Meter made by Unihedron
https://codeberg.org/alteholz/PySQM.git

PySQM INAF - PySQM version for the SQM-LE (Unihedron) adapted for DARKERSKY4CE Photometer Network
https://www.ict.inaf.it/gitlab/dario.barghini/pysqm_darkersky4ce

RClone - Download page of the RClone tool
https://rclone.org/downloads/

SQM-LE - An Ethernet-enabled SQM model for night-time sky brightness monitoring
https://www.unihedron.com/projects/sqm-le/

Unihedron - Unihedron main webpage
https://www.unihedron.com/