Setup & Operation

Table of Contents

1. **Introduction**
   The following document contains information about setting up and using the Takano desktop application. The app is freely available on http://iotsprint.com/takano_desktop.zip. It is an easy-to-use, intuitive and useful application to seamlessly connect and exchange data with the Takano cloud-based platform. Before delving into the app details, it should be noted that the user ought to already have setup both the hardware board(s), the Takano cloud <u>and</u> the mobile app before going for this desktop app. In addition he should have also saved the setup to the Takano server from the mobile app using the "Save To Cloud" button. The desktop app thus requires the user to download this setup from the Takano server and does not edit or modify it in any way. It offers a mean to archive the data sent by the hardware board(s) by saving them on a daily basis in Excel comma separated value files (.csv), as well as providing simple line graph charts to display the hardware data. It is an easy and convenient way of storing hardware data in tables, which could be useful in some projects. Another difference with the mobile app is the fact that it cannot receive emails and push notifications. The desktop app has been compiled to run on most popular operating systems including Windows, Mac and Linux. The only requirement is to have the Java Runtime Environment already installed (recommended version 8 or above). This can be downloaded from https://www.java.com/en/download/. Please note that this app does not run on single-board-computers like the Raspberry pi because it requires special JavaFX components which do not easily run on Raspbian operating system, even when installed separately. For that reason we have written a scaled-down Python script, also freely available at http://iotsprint.com/takano_sbc.py, that can run on any Python-installed computer with any OS, even without a screen and a keyboard, just to save the data on .CSV files. The first step is to connect to www.iotsprint.com website from his PC using any modern browser. Then he presses on the "Takano" keyword in the top-most menu. The browser will automatically scroll down to the "Takano" section. In this section there is a "Desktop" button which he should press in order to start the "takano_desktop.zip" download. After the download and saving is through, he unzips it to any directory he wants on his PC. The user is then presented with the takano_desktop directory that contains all the necessary assets to run it.

2. **Setup & Run the Desktop App**
   <u>Windows</u>:
   Inside this newly-created takano_desktop directory, double-click on Takano.exe or Takano.jar to launch the app.
   <u>Mac/Linux</u>:
   Open a terminal. Enter a command-line to access the takano_desktop directory (supposing it was saved on the desktop): cd/desktop/ takano_desktop. Then another command to launch the .jar file: java -jar takano.jar

When launching takano app for the first time, the user is presented with a blank page with a "geared" button on the lower right corner as in Fig 1. The user has to press on it in order to move to the setup page as in Fig 2. A constant internet connection is required for a successful setup.
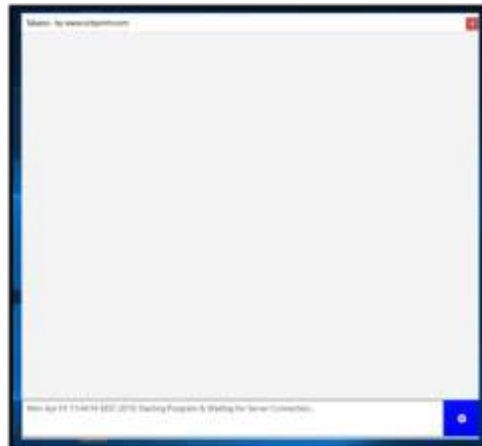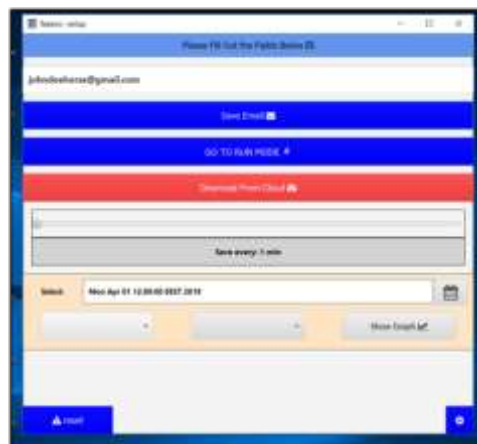

Fig 1.

Fig 2.

Setup is straight forward as it requires just two steps. The first step is to enter the email provided to the takano platform setup in the "Enter Email Here" field, for example: johndoehorse@gmail.com. Then he saves the email by pressing the "Save Email" button. The next step is to press on the "Download From Cloud" button. The previous email field now changes to prompt the user to "Enter Takano Password Here". This is the administrator password he has already entered on the Takano platform. After entering it he presses on the "SAVE TAKANO PASSWORD" in red font. If the entered email and password match those on the Takano server, then the setup data will be downloaded and saved on the PC. A pop-up message on the bottom of the form will acknowledge it. At this stage the user can press on "GO TO RUN MODE" to be directed to the main form where he can watch the hardware variables

being updated in real time from the Takano cloud and also send commands back to the hardware if included in the setup, just like in the mobile app.

3. **Display data in the Desktop app**

Back to the Setup form, there is a slider in the middle of the form that allows the user to specify the frequency saving of the data. The slider values are: 1 minute, 5 minutes, 10 minutes, 15 minutes, 20 minutes, 25 minutes, 30 minutes, 1 hour, 2 hours, 3 hours and 6 hours. The user can adjust it to his needs. There is no need to do any saving as the effect is immediate. The app has to be constantly connected online for the saving to take effect. The main form also needs to be launched (and hence the setup form should be closed). As stated in the introduction, the data is saved in .csv files. In the created takano_desktop directory, the saved files are stored under Objects>Takano>Archive>year>month>day subdirectories. Each hardware has the .csv files named after it. And each variable sent through the Takano cloud occupies one column in that file. Suppose for example there is a hardware named "Mountain_house", then the daily created file will for instance be: takano_desktop>Objects>Takano>Archive>2024>Jan>25>Mountain_house.csv.

When the clock ticks after midnight, a new day and a new .csv file will be created: takano_desktop>Objects>Takano>Archive>2024>Jan>26>Mountain_house.csv.

When a new month quicks in, a new month directory will be created: takano_desktop>Objects>Takano>Archive>2024>Jan>1>Mountain_house.csv.

The same logic also applies when a new year starts… Since the data is saved in .csv files, Excel can of course be used to view the worksheet, draw various graphical representations and analysis if needed. But the Takano desktop app also provides a simple and quick way of visualizing a particular hardware variable: from the Setup form, the user simply chooses the date by clicking on the calendar icon, specifies the hardware and variable names from the two provided combo boxes and then presses on "Show Graph" button. A new form will be displayed that shows a line chart with the data for that particular day as seen in Fig 3. Pressing on the mouse left button anywhere on the chart will show the x value which is the time, the y value which is the Voltage V in this case, the maximum V on the chart (MX=132), the average V (AV=118.875) and the minimum V (MI=101). One last remark: on some operating systems like Windows 10, it is preferable to move from the main form to the setup form in order to stop saving the data if the user wishes to open Excel, as Windows will not allow the app to open Excel if it is already opened by the user. That will not cause the app to crash but saving will be paused until the user closes Excel.

Fig 3.

4. SBC Python script

The paragraph contains information about setting up and using the Takano single-board-computer (SBC) application. The app is a Python script file (.py) that can work on any computer running the latest Python 3 version, which freely downloads from https://www.python.org/downloads/. The script is downloaded from http://iotsprint.com/takano_sbc.py. Although the Takano desktop app we previously depicted in this document works for all major operating systems (Linux, Windows and Mac), it does not specifically run on single-board-computers like the Raspberry pi due to their Java Run Time environment being different from the one required to run on larger computers such as laptops or desktops. For that reason this SBC Python script was written. Compared to the desktop app, it is a scaled-down, non-graphical user interface i.e. that does not require a screen or keyboard to run. Users with some experience in SBCs and Linux can launch it at specific times using "Crontab" for example. That makes the SBC Python script suitable to run on any Raspberry pi, even one running remotely on a network with internet access. It is a very straight forward and succinct script that connects to and saves the data from the Takano cloud-based platform. It could also form the basis of a user-customized app or program. Before viewing the app flow, it should be noted that the user ought to already have setup the hardware board(s) and the Takano cloud before running this Python script. This script does not edit or modify the cloud data in any way. It offers a simple way of archiving the data sent by the hardware board(s) by saving them on a daily basis in comma separated value files (.csv), just by sending a refresh command to the cloud with the user registration key that was obtained from the Takano platform during registration. Open-source spreadsheet Software like LibreOffice that ships with Raspberry pi can be used to view those tables. Another difference with the mobile app is the fact that it cannot receive emails and push notifications. To view and run this script, you can use the IDLE development environment that downloads with the official Python and is pre-installed in Raspberry pi and other SBCs. The script is few tens of lines, please go over it and read the comments as you have to manually enter the user registration number(s) and the name(s) of the hardware boards you want to save data from.

5. Contact information

For more information or comments, please do not hesitate to contact the Takano technical team by email at: info@iotsprint.com
or by phone at: +961 1 246 933
We are available from 8:00AM (GMT+3) to 6:00PM (GMT+3).