### **http://docs.blynk.cc/#widgets-displays-superchart**

### 

### **SuperChart**

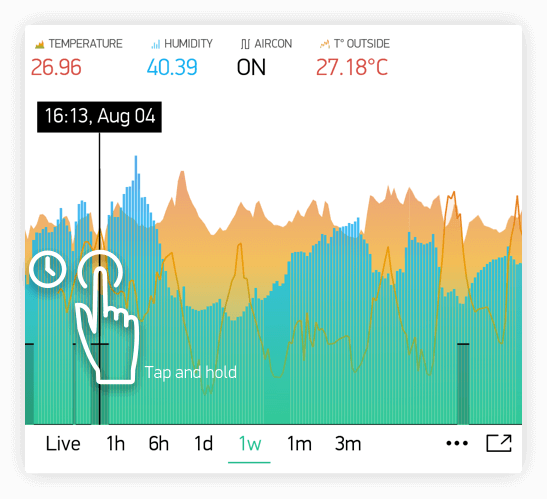
SuperChart is used to visualise live and historical data. You can use it for sensor data, for binary event logging and more.

To use SuperChart widget you would need to push the data from the hardware with the desired interval by using timers.

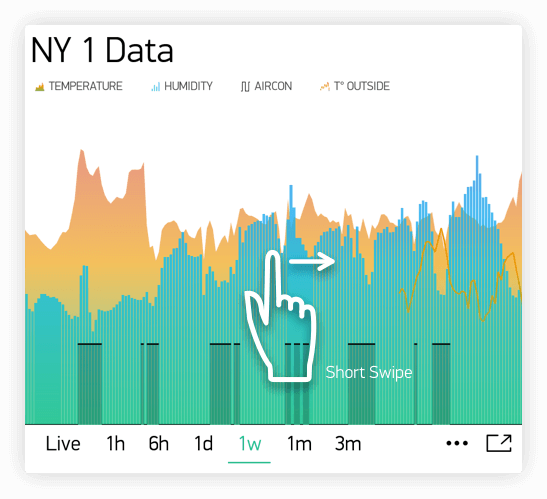
[Here is](https://examples.blynk.cc/?board=ESP8266&shield=ESP8266%20WiFi&example=GettingStarted%2FPushData) a basic example for data pushing.

#### **Interactions:**

* **Switch between time ranges and Live mode**
* Tap time ranges at the bottom of the widget to change time ranges
* **Tap Legend Elements** to show or hide datastreams
* **Tap’n’hold to view timestamp and corresponding values**

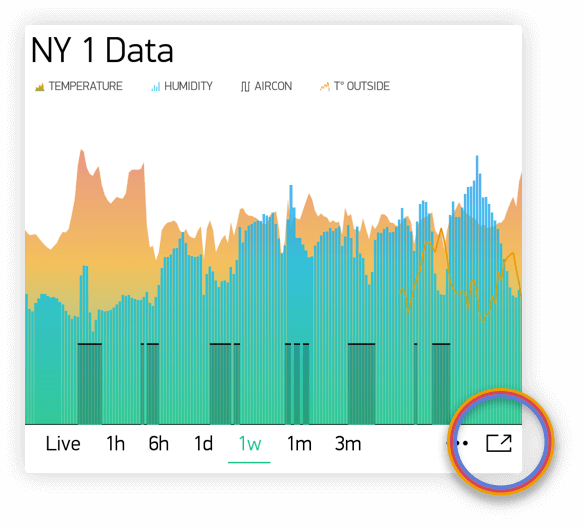
****

* **Quick swipe from left to right to reveal previous data**

****

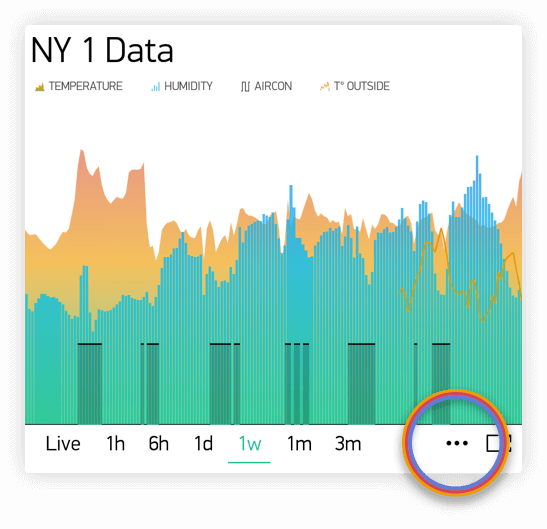
Then you can then scroll data back and forward within the given time range.

* **Full Screen Mode**
* Press this button to open Full Screen view in landscape orientation:



Simply rotate the phone back to portrait mode. Chart should rotate automagically. In full screen view you will see X (time) and multiple Y scales. Full Screen Mode can be disabled from widget Settings.

* **Menu Button**
* Menu button will open additional functions:
  + Export to CSV
  + Erase Data on the server

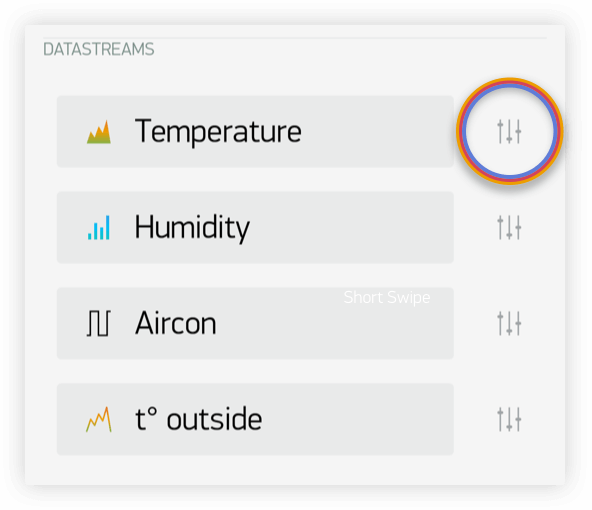


#### **SuperChart (History Graph) Settings:**

* **Chart Title**
* **Title Font Size** You have a choice of 3 font sizes
* **Title Alignment** Choose chart title alignment. This setting also affects Title and Legend position on the Widget.
* **Datastreams** - add datastreams (read below how to configure datastreams)
* **Show/Hide Title**
* **Show/Hide Legend**

#### **Datastream Settings**

Widget supports up to 4 Datastreams. Press Datastream Settings Icon to open Datastream Settings.



**Design:** Choose available types of Chart:

* Line
* Area
* Bar
* Binary (anchor LINK to binary)

**Color:** Choose solid colors or gradients

**Source and input:** You can use 3 types of Data source:

**1. Virtual Pin** Choose the desired Device and Virtual Pin to read the data from.

**2. Tags** SuperChart can aggregate data from multiple devices using built-in aggregation functions. For example, if you have 10 Temperature sensors sending temperature with the given period, you can plot average value from 10 sensors on the widget.

To use Tags:

* [**Add Tag**](http://docs.blynk.cc/#blynk-main-operations-control-of-multiple-devices-tags) to every device you want to aggregate data from.
* **Push data to the same Virtual Pin** on every device. (e.g. Blynk.virtualWrite (V0, temperature);)
* **Choose Tag as a source** in SuperChart Widget and use the pin where the data is coming to (e.g V0)

**Functions available:**

* **SUM**, will summarize all incoming values to the specified Virtual Pin across all devices tagged with the chosen tag
* **AVG**, will plot average value
* **MED**, will find a median value
* **MIN**, will plot minimum value
* **MAX** will plot minimum value

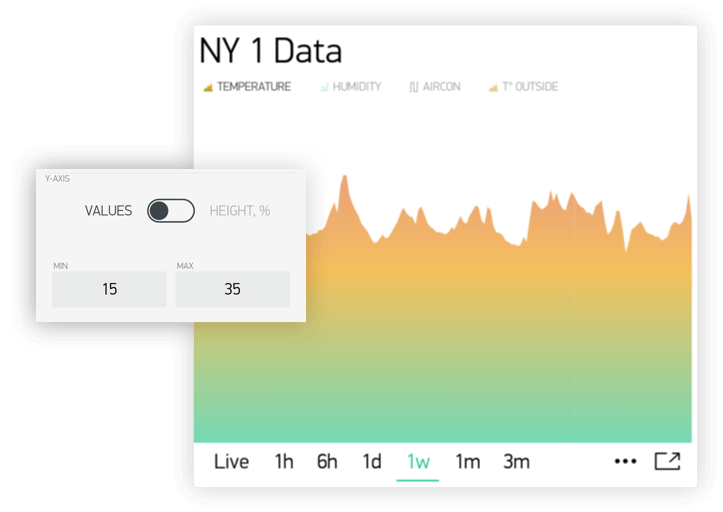
**☝️ IMPORTANT: Tags are not working in Live Mode.**

1. [**Device Selector**](http://docs.blynk.cc/#widgets-time-input-device-selector) If you add Device Selector Widget to your project, you can use it as a source for SuperChart. In this case, when you change the device in Device Selector, chart will be updated accordingly

**Y-Axis Settings**

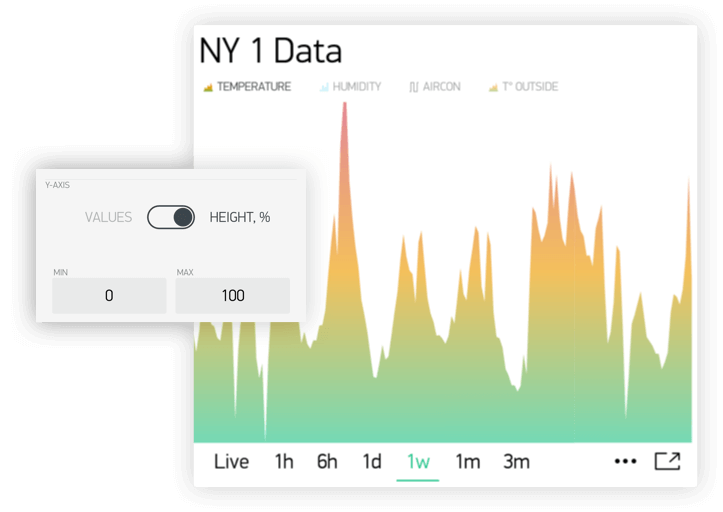
There are two modes of how to scale data along the Y axis

1. **Values**
2. When this mode is selected, Y scale will be set to the values you choose. For example, if your hardware sends data with values varying from -100 to 100, you can set the chart to this values and data will be rendered correctly.



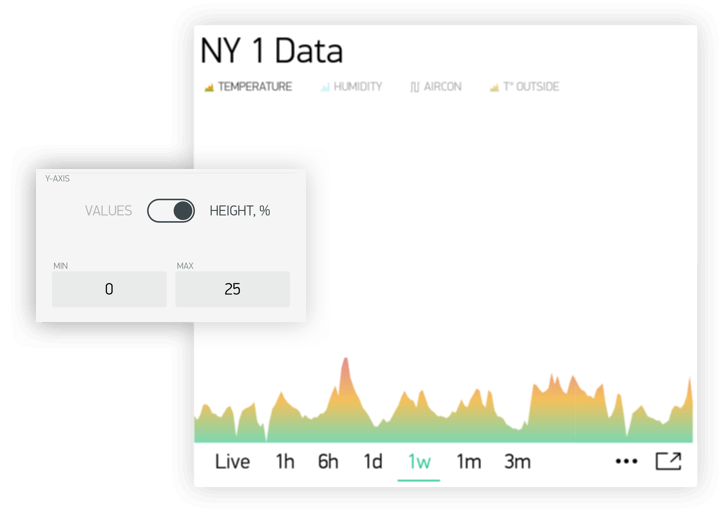
You may also want to visualize the data within some specific range. Let’s say incoming data has values in the range of 0-55, but you would like to see only values in the range 30-50. You can set it up and if values are out of Y scale you configured, chart will be cropped

1. **% of Height**
2. This option allows you to auto-scale incoming data on the widget and position it the way you want. In this mode, you set up the percentage of widget height on the screen, from 0% to 100%.

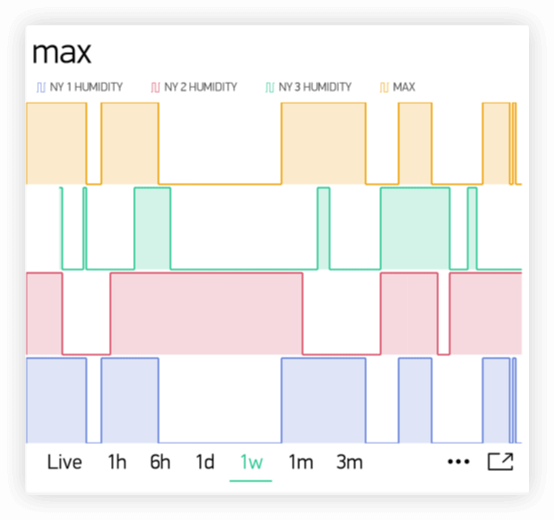


If you set 0-100%, in fact it’s a full auto-scale. No matter in which range the data is coming,

it will be always scaled to the whole height of the widget.

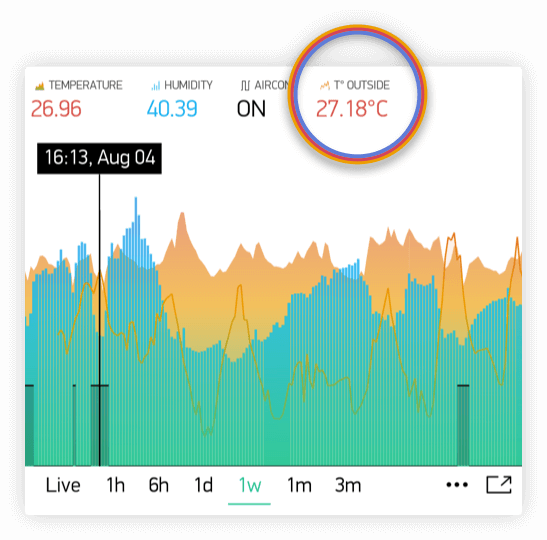
If you set it to 0-25%, then this chart will only be rendered on 1/4 of the widget height: 

This setting is very valuable for **Binary Chart** or for visualizing a few datastreams on the same chart in a different way.



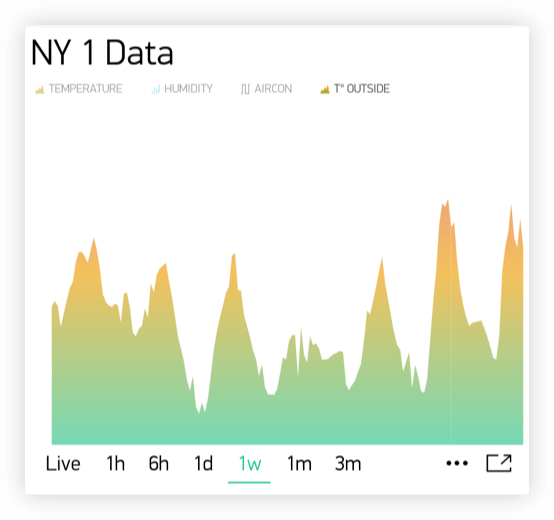
**Suffix:**

Here you can specify a suffix that will be shown during the Tap’n’hold

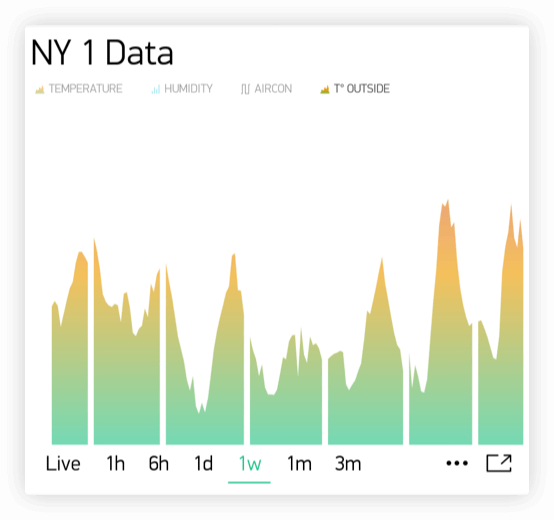


**Connect Missing Data Points**

If this switch is ON, then SuperChart will connect all the dots even if there was no data



If it’s set to OFF, then you will see gaps in case there was no data.



**Binary Chart Settings**

This type of chart is useful to plot binary data, for example when unit was ON or OFF, or when motion was detected or when certain threshold was reached.

You need to specify a **FLIP** point, which is the point where incoming data will be turned into TRUE or FALSE state.

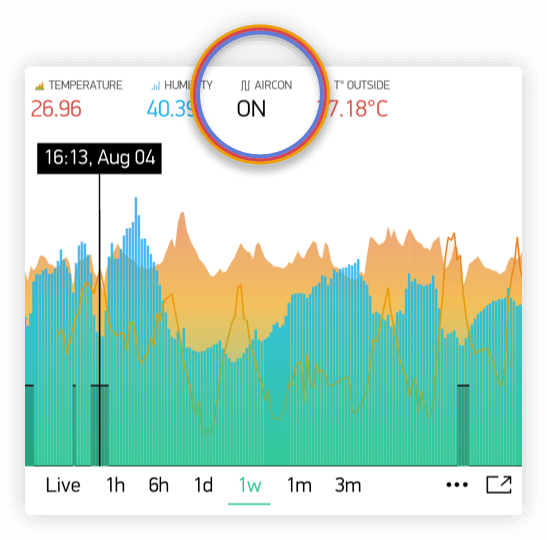
For example, you send the data in the range of 0 to 1023. If you set 512 as a **FLIP** point, then everything above 512 (excluding 512) will be recorded as TRUE, any value below 512 (including 512) will be FALSE.

Another example, if you send 0 and 1 and set 0 as a **FLIP** point, then 1 will be TRUE, 0 will be FALSE

**State Labels:**

Here you can specify how TRUE/FALSE should be shown in Tap’n’Hold mode.

For example, you can set to TRUE to “Equipment ON” label, FALSE to “Equipment OFF”.



Superchart supports currently 2 types of granularity:

* Minute granularity - 1h, 6h, 1d;
* Hour granularity - 1w, 1m, 3m;

This means that minimum chart update interval is 1 minute for 1h, 6h, 1d periods. 1 hour for 1w, 1m and 3m periods. As Blynk Cloud is free to use we have a limit on how many data you can store. At the moment Blynk Cloud accepts 1 message per minute per pin. In case you send your data more frequently your values will be averaged. For example, in case you send value 10 at 12:12:05 and than again 12 at 12:12:45 as result in chart you’ll see value 11 for 12:12.

In order to see data in chart you need to use either widgets with “Frequency reading” interval (in that case your app should be open and running) or you can use Blynk.virtualWrite on hardware side. Every Blynk.virtualWrite command is stored on server automatically. In that case you don’t need application to be up and running.